FIG. 1

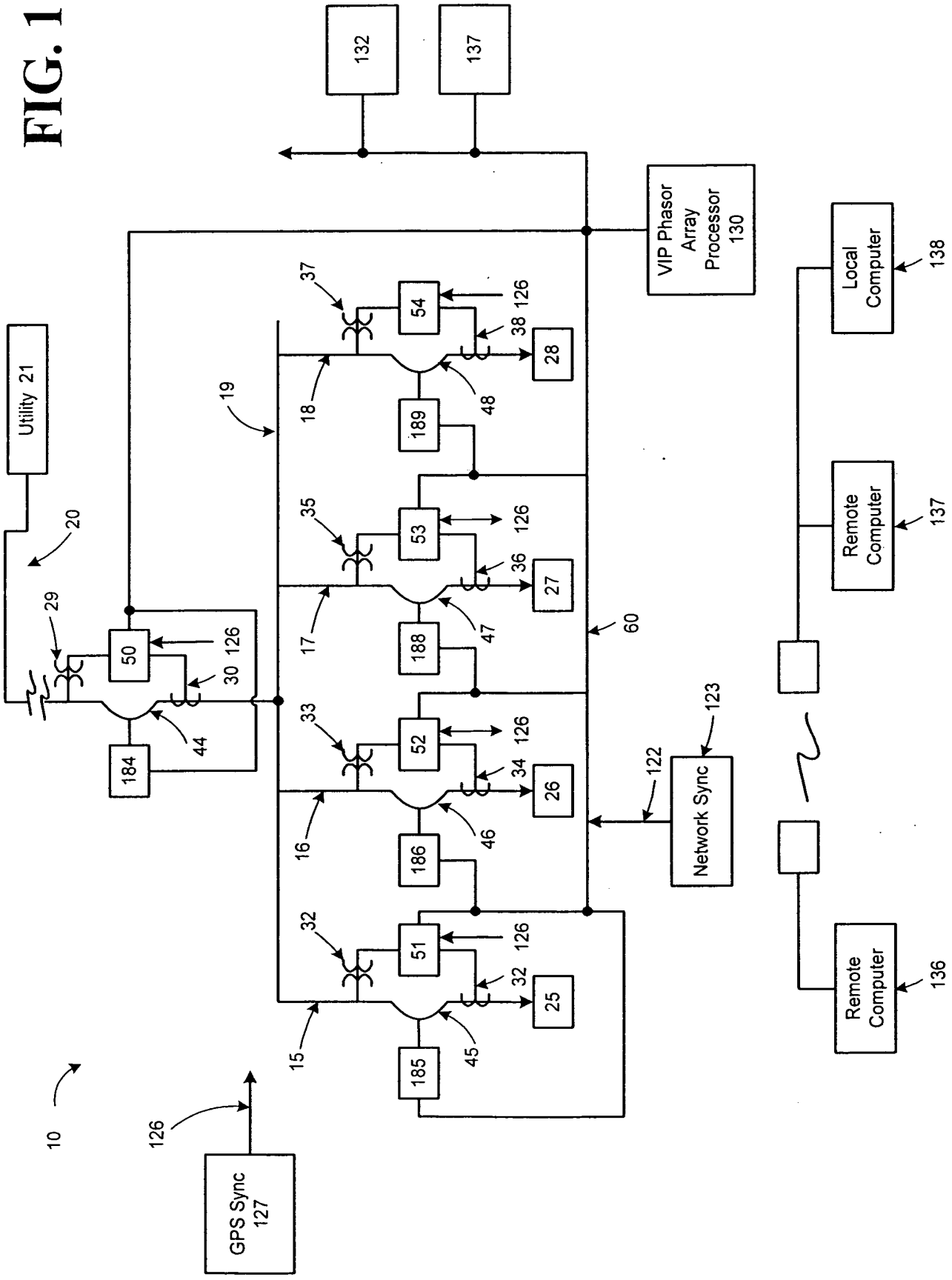

132
137
VIP Phasor Array Processor 130
Local Computer
138
Remote Computer
137
Remote Computer
136
Utility 21
20
19
18
17
16
15
10
60
37
35
33
32
54
53
52
51
38 126
36
34
32
28
27
26
25
189
188
186
185
48
47
46
45
29
50
30 126
44
184
Network Sync
123
122
GPS Sync 127
126

FIG. 2

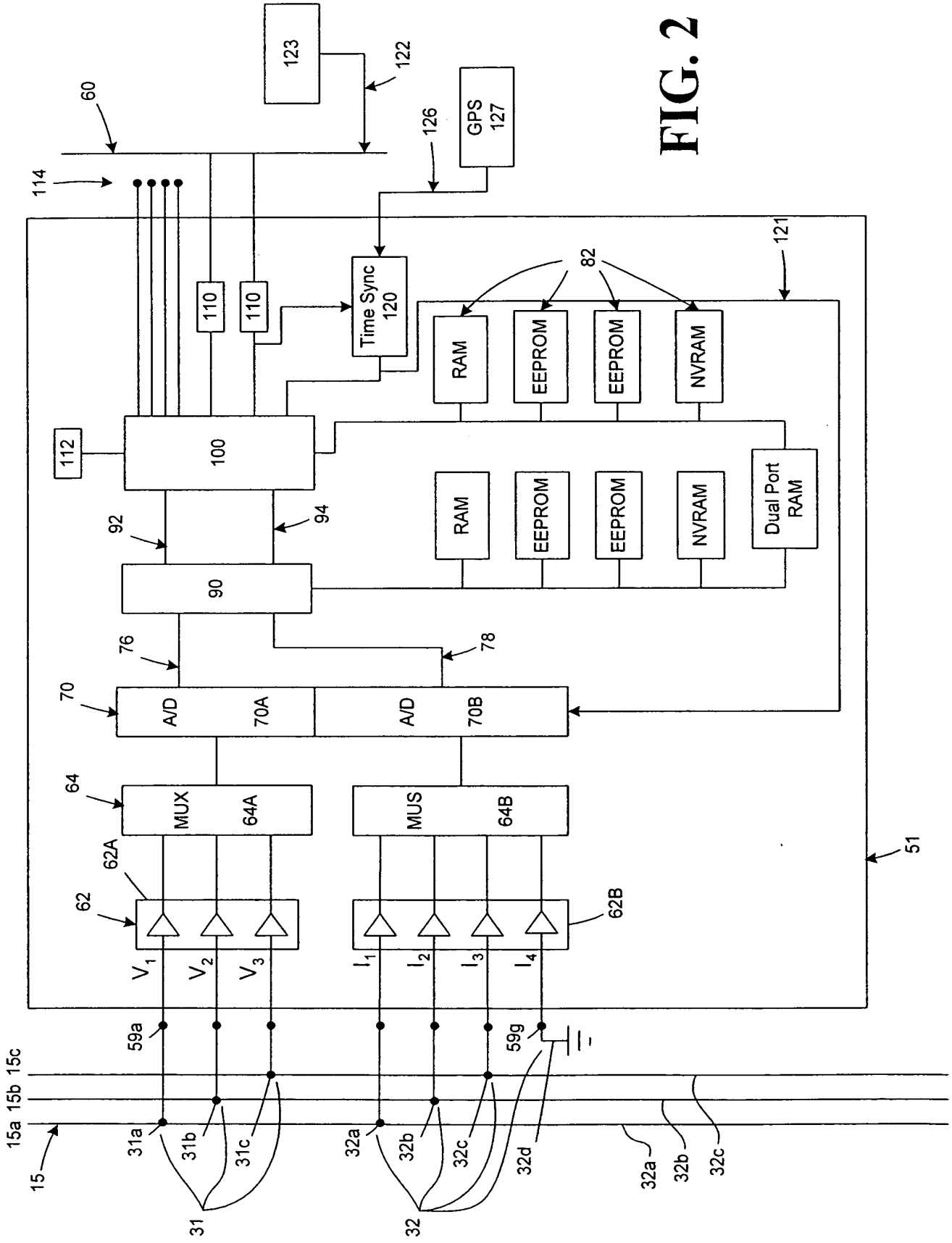

123
122
60
126
GPS 127
114
121
82
110
110
Time Sync 120
RAM
EEPROM
EEPROM
NVRAM
112
100
RAM
EEPROM
EEPROM
NVRAM
Dual Port RAM
92
94
90
76
78
70
A/D
70A
A/D
70B
64
MUX
64A
MUS
64B
62A
62
62B
51
V_1
V_2
V_3
I_1
I_2
I_3
I_4
59a
59g
15c
15b
15a
15
31a
31b
31c
31
32a
32b
32c
32d
32
32a
32b
32c

FIG. 3

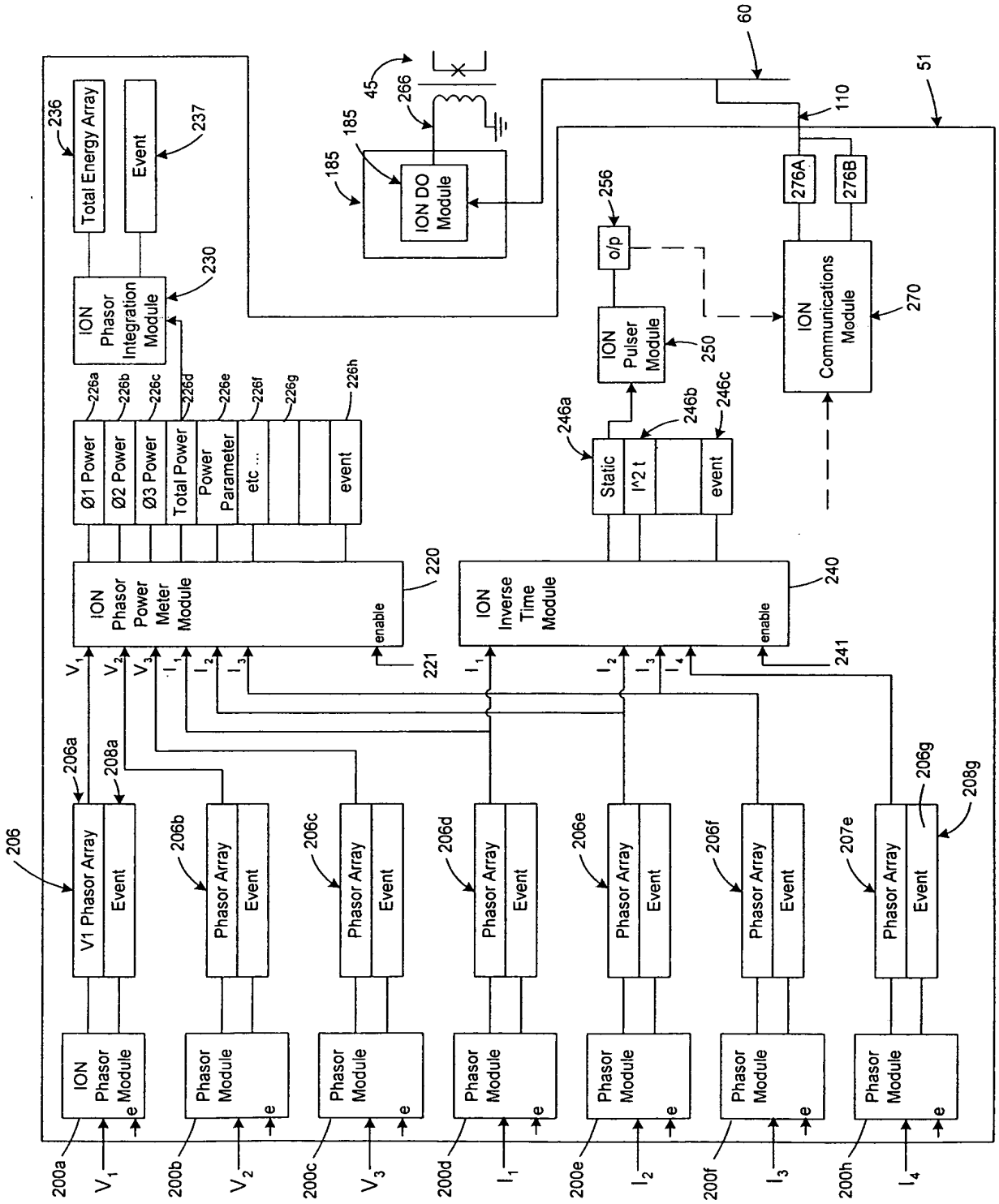

Phasion ION Moudle Diagram
ION Phasor Module
Phasor Module
V1 Phasor Array
Phasor Array
Event
ION Phasor Power Meter Module
ION Phasor Integration Module
ION Inverse Time Module
ION Pulser Module
ION Communications Module
ION DO Module
Total Energy Array
Ø1 Power
Ø2 Power
Ø3 Power
Total Power
Power Parameter
etc ...
Static
I^2 t
o/p
enable
276A
276B
200a
200b
200c
200d
200e
200f
200h
206
206a
206b
206c
206d
206e
206f
207e
206g
208a
208g
220
221
230
236
237
240
241
250
256
270
185
45
266
60
51
110

FIG. 4

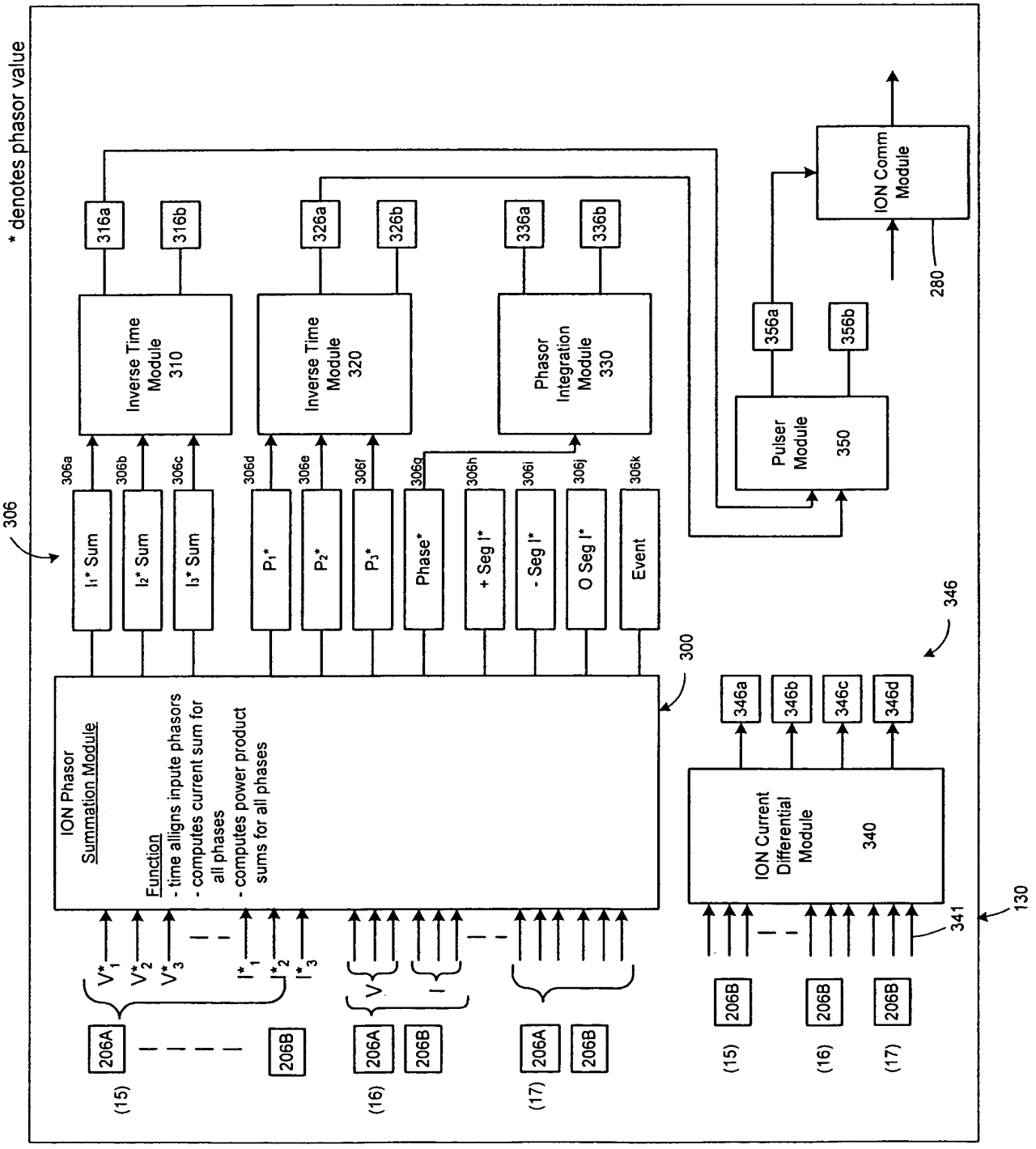

* denotes phasor value
306
316a
316b
326a
326b
336a
336b
Inverse Time Module 310
Inverse Time Module 320
Phasor Integration Module 330
ION Comm Module
280
356a
356b
Pulser Module 350
306a
306b
306c
306d
306e
306f
306g
306h
306i
306j
306k
I1* Sum
I2* Sum
I3* Sum
P1*
P2*
P3*
Phase*
+ Seg I*
- Seg I*
O Seg I*
Event
300
ION Phasor Summation Module
Function
- time aligns input phasors
- computes current sum for all phases
- computes power product sums for all phases
346
346a
346b
346c
346d
ION Current Differential Module 340
341
130
V*1
V*2
V*3
I*1
I*2
I*3
V
I
206A
206B
(15)
(16)
(17)

FIG. 5

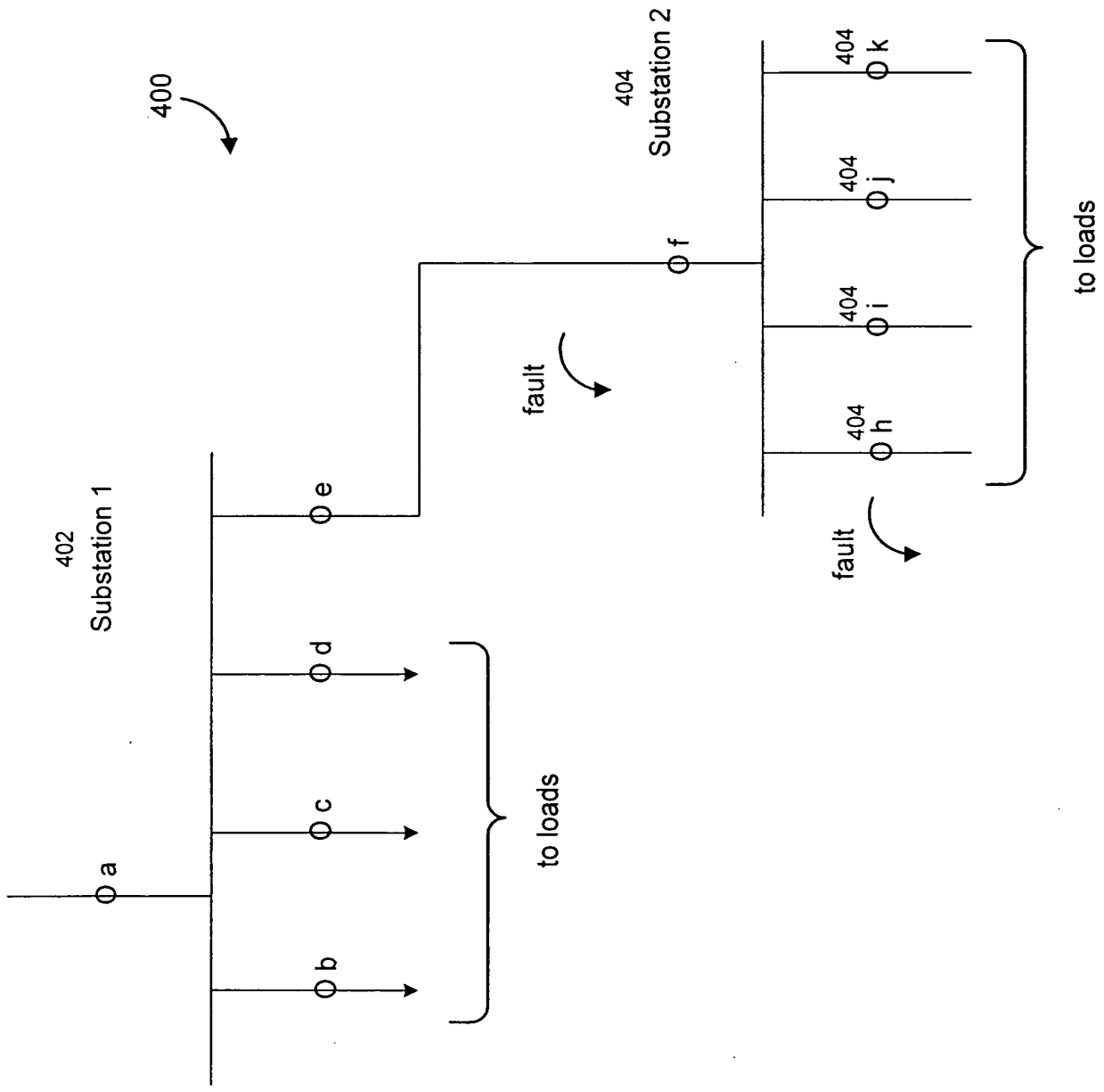

400
402
Substation 1
404
Substation 2
a
b
c
d
e
f
404 h
404 i
404 j
404 k
fault
fault
to loads
to loads

FIG. 6

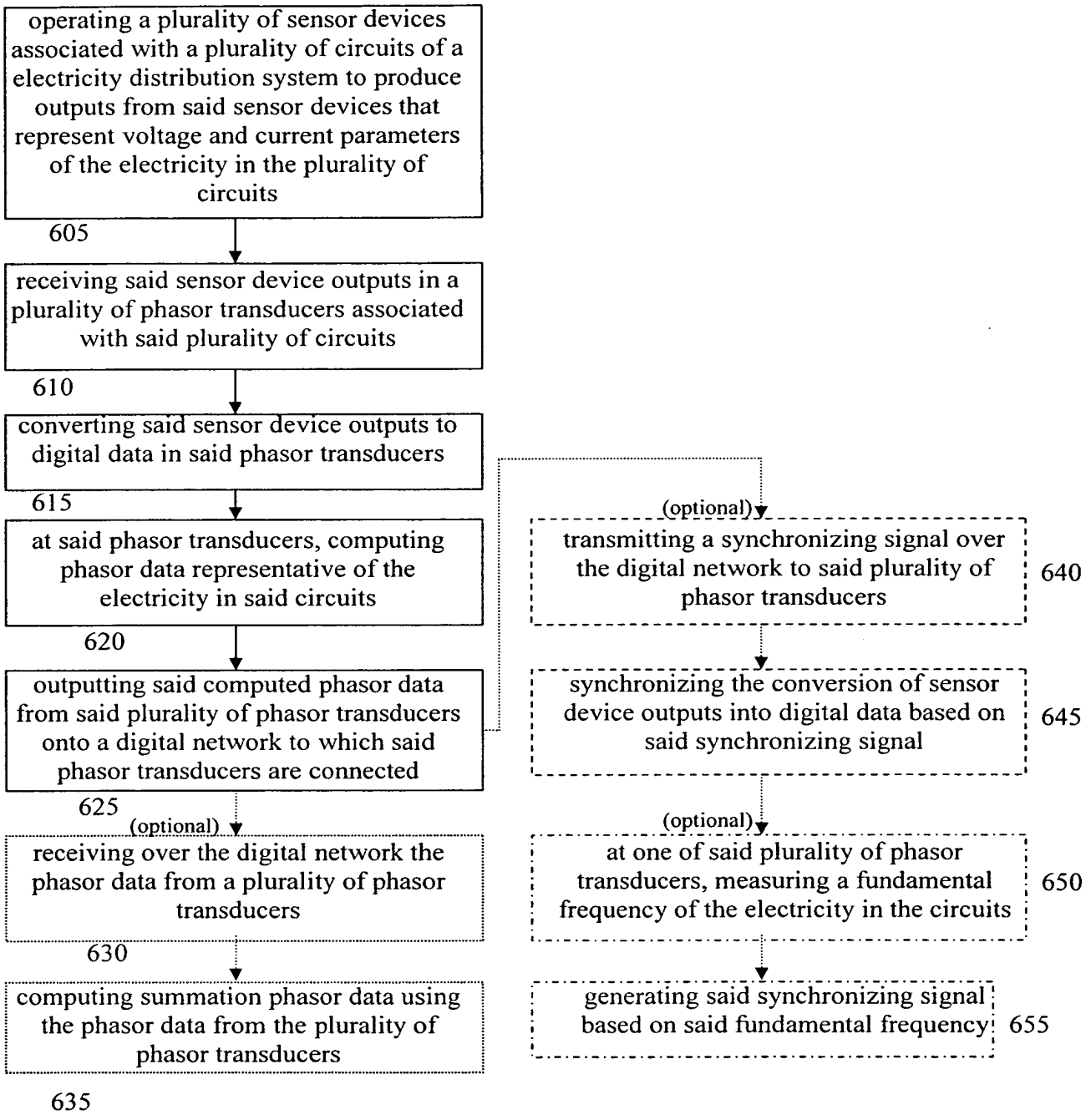

operating a plurality of sensor devices associated with a plurality of circuits of a electricity distribution system to produce outputs from said sensor devices that represent voltage and current parameters of the electricity in the plurality of circuits
605
receiving said sensor device outputs in a plurality of phasor transducers associated with said plurality of circuits
610
converting said sensor device outputs to digital data in said phasor transducers
615
at said phasor transducers, computing phasor data representative of the electricity in said circuits
620
outputting said computed phasor data from said plurality of phasor transducers onto a digital network to which said phasor transducers are connected
625
(optional)
receiving over the digital network the phasor data from a plurality of phasor transducers
630
computing summation phasor data using the phasor data from the plurality of phasor transducers
635
(optional)
transmitting a synchronizing signal over the digital network to said plurality of phasor transducers
640
synchronizing the conversion of sensor device outputs into digital data based on said synchronizing signal
645
(optional)
at one of said plurality of phasor transducers, measuring a fundamental frequency of the electricity in the circuits
650
generating said synchronizing signal based on said fundamental frequency
655

FIG. 7

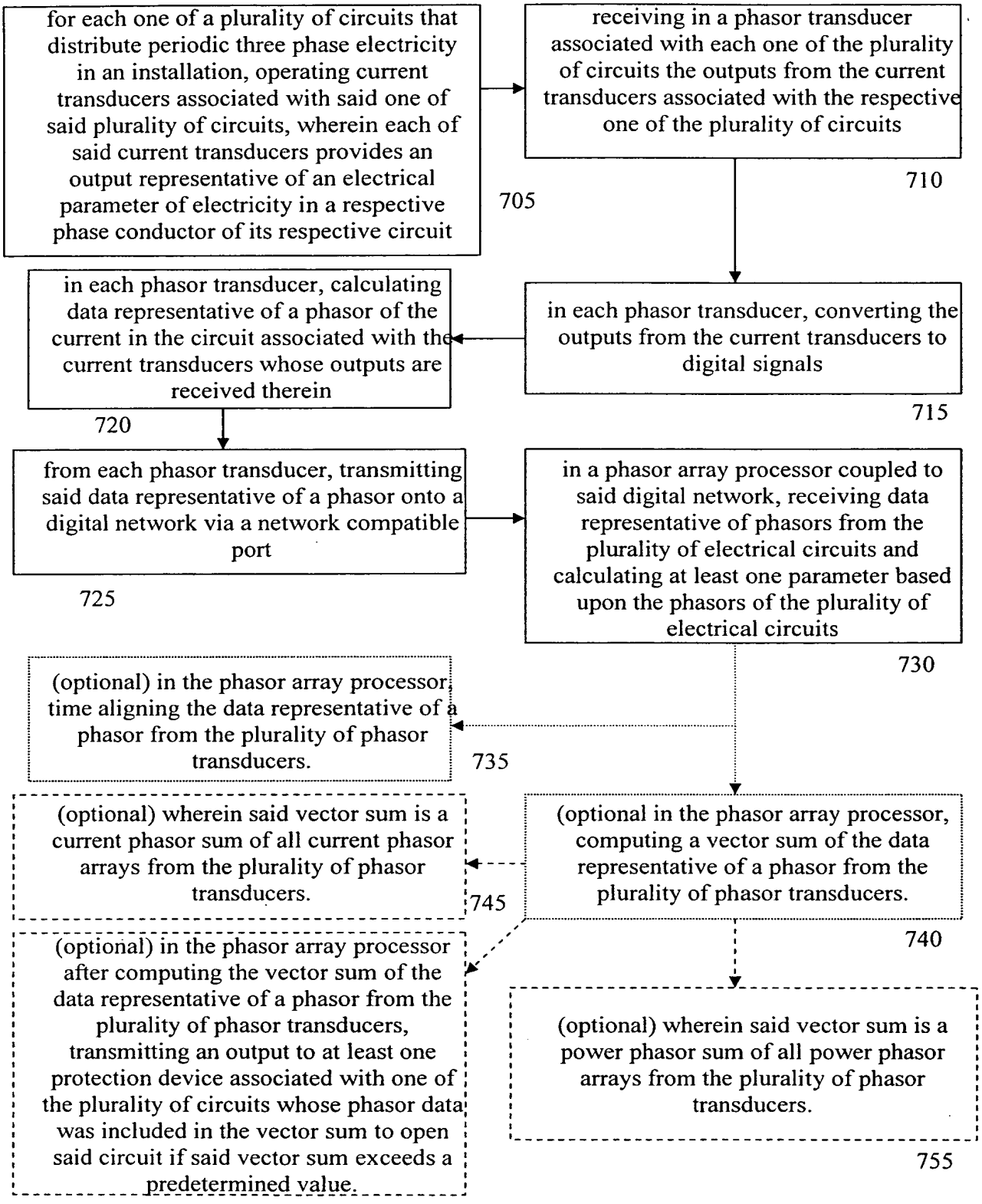

for each one of a plurality of circuits that distribute periodic three phase electricity in an installation, operating current transducers associated with said one of said plurality of circuits, wherein each of said current transducers provides an output representative of an electrical parameter of electricity in a respective phase conductor of its respective circuit
705
receiving in a phasor transducer associated with each one of the plurality of circuits the outputs from the current transducers associated with the respective one of the plurality of circuits
710
in each phasor transducer, converting the outputs from the current transducers to digital signals
715
in each phasor transducer, calculating data representative of a phasor of the current in the circuit associated with the current transducers whose outputs are received therein
720
from each phasor transducer, transmitting said data representative of a phasor onto a digital network via a network compatible port
725
in a phasor array processor coupled to said digital network, receiving data representative of phasors from the plurality of electrical circuits and calculating at least one parameter based upon the phasors of the plurality of electrical circuits
730
(optional) in the phasor array processor time aligning the data representative of a phasor from the plurality of phasor transducers.
735
(optional in the phasor array processor, computing a vector sum of the data representative of a phasor from the plurality of phasor transducers.
740
(optional) wherein said vector sum is a current phasor sum of all current phasor arrays from the plurality of phasor transducers.
745
(optional) in the phasor array processor after computing the vector sum of the data representative of a phasor from the plurality of phasor transducers, transmitting an output to at least one protection device associated with one of the plurality of circuits whose phasor data was included in the vector sum to open said circuit if said vector sum exceeds a predetermined value.
750
(optional) wherein said vector sum is a power phasor sum of all power phasor arrays from the plurality of phasor transducers.
755

FIG. 8

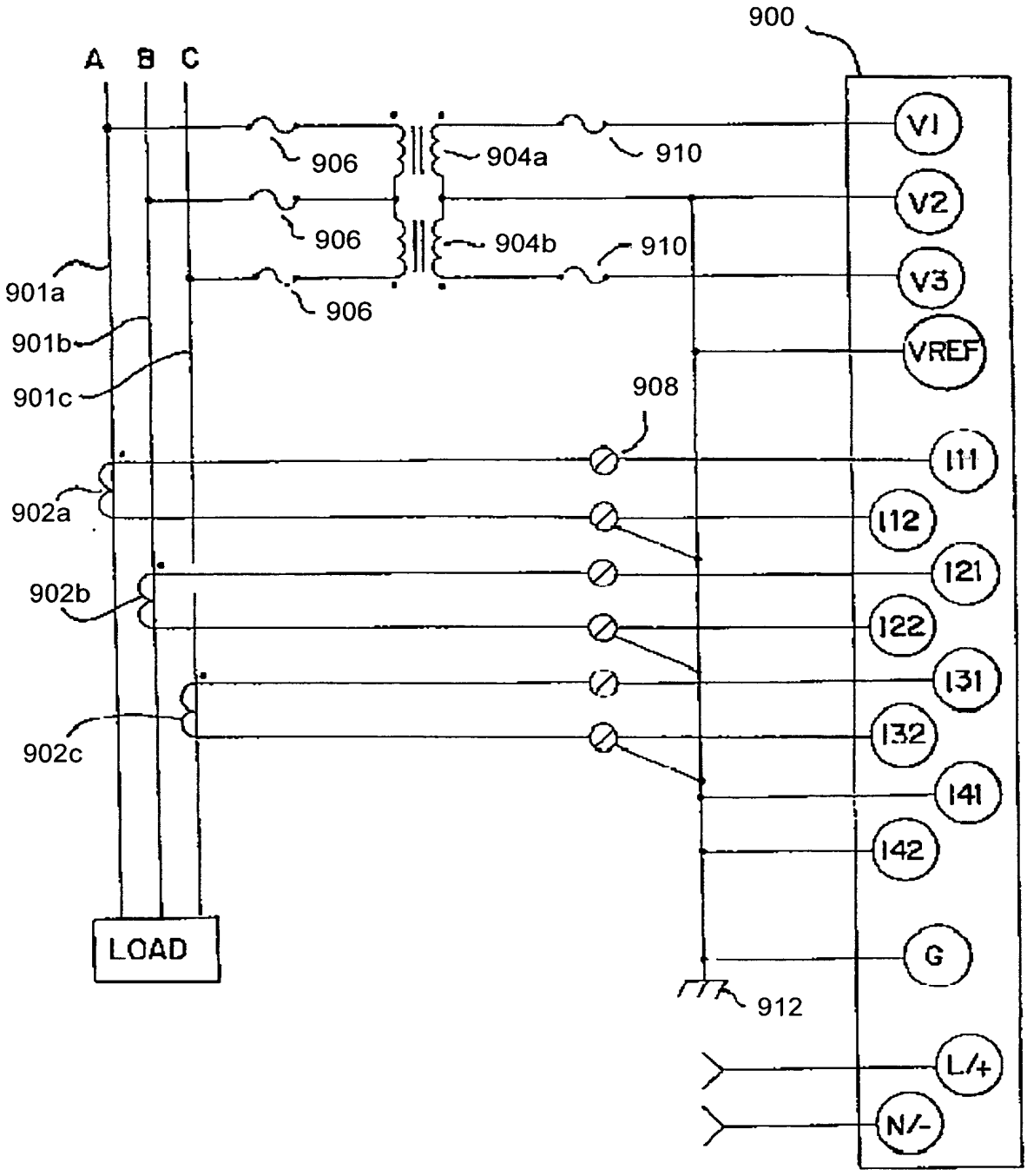

900
A
B
C
906
904a
910
906
904b
910
906
901a
901b
901c
908
902a
902b
902c
V1
V2
V3
VREF
I11
I12
I21
I22
I31
I32
I41
I42
G
L/+
N/-
912
LOAD

FIG. 9

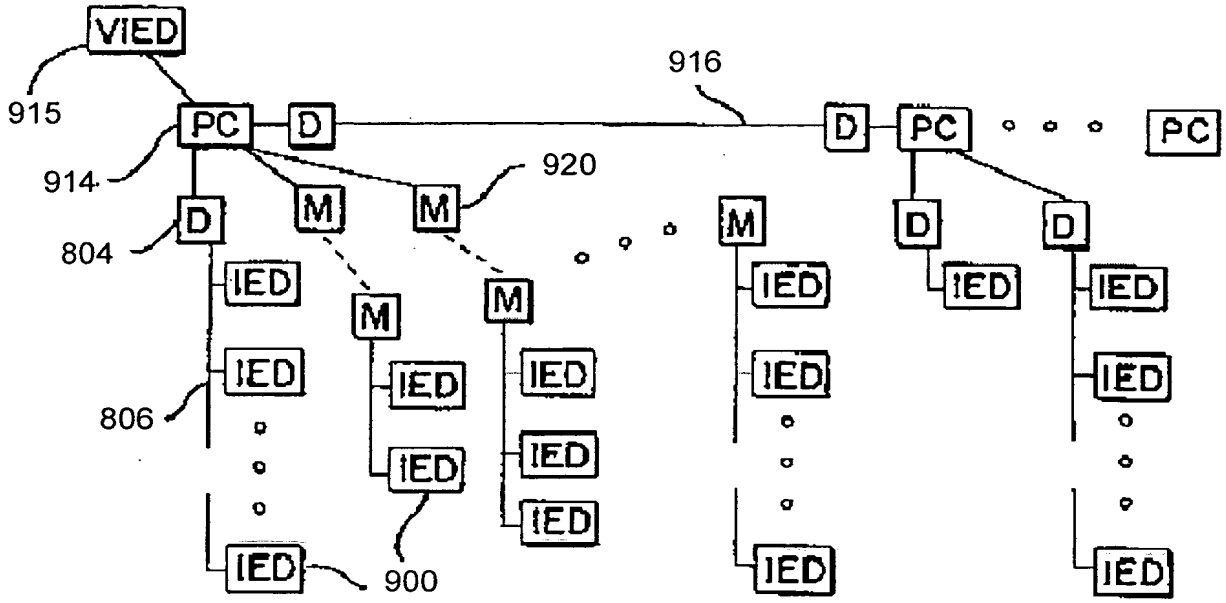


FIG. 11

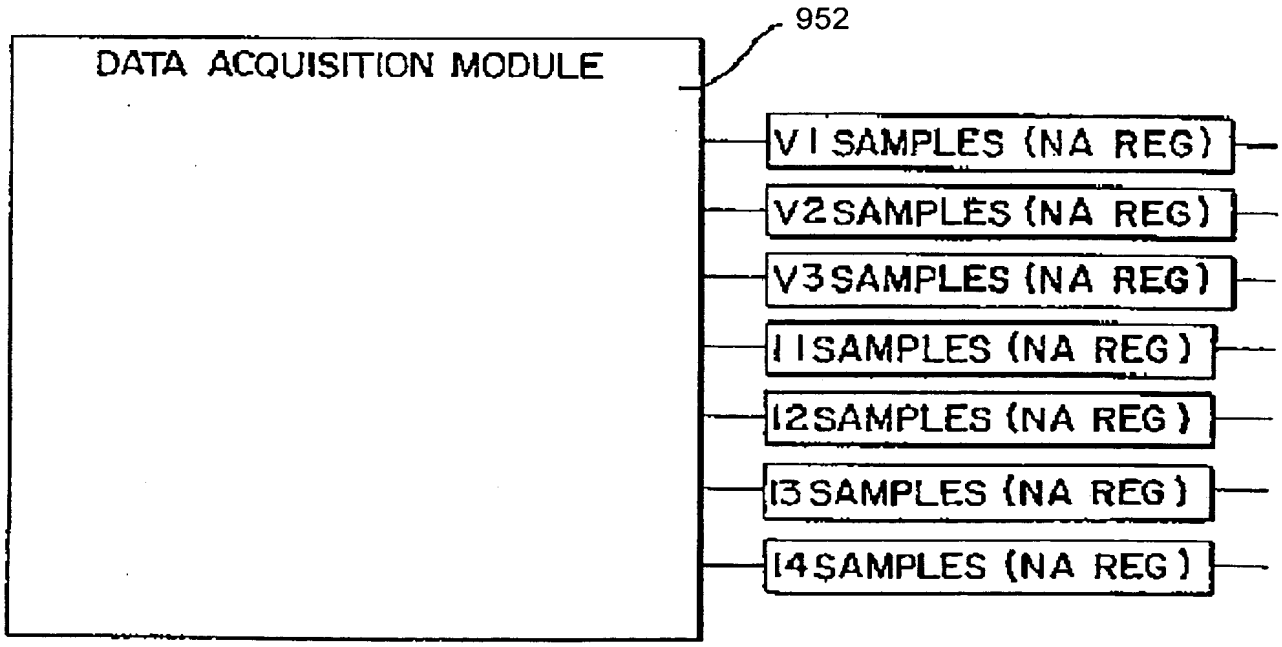
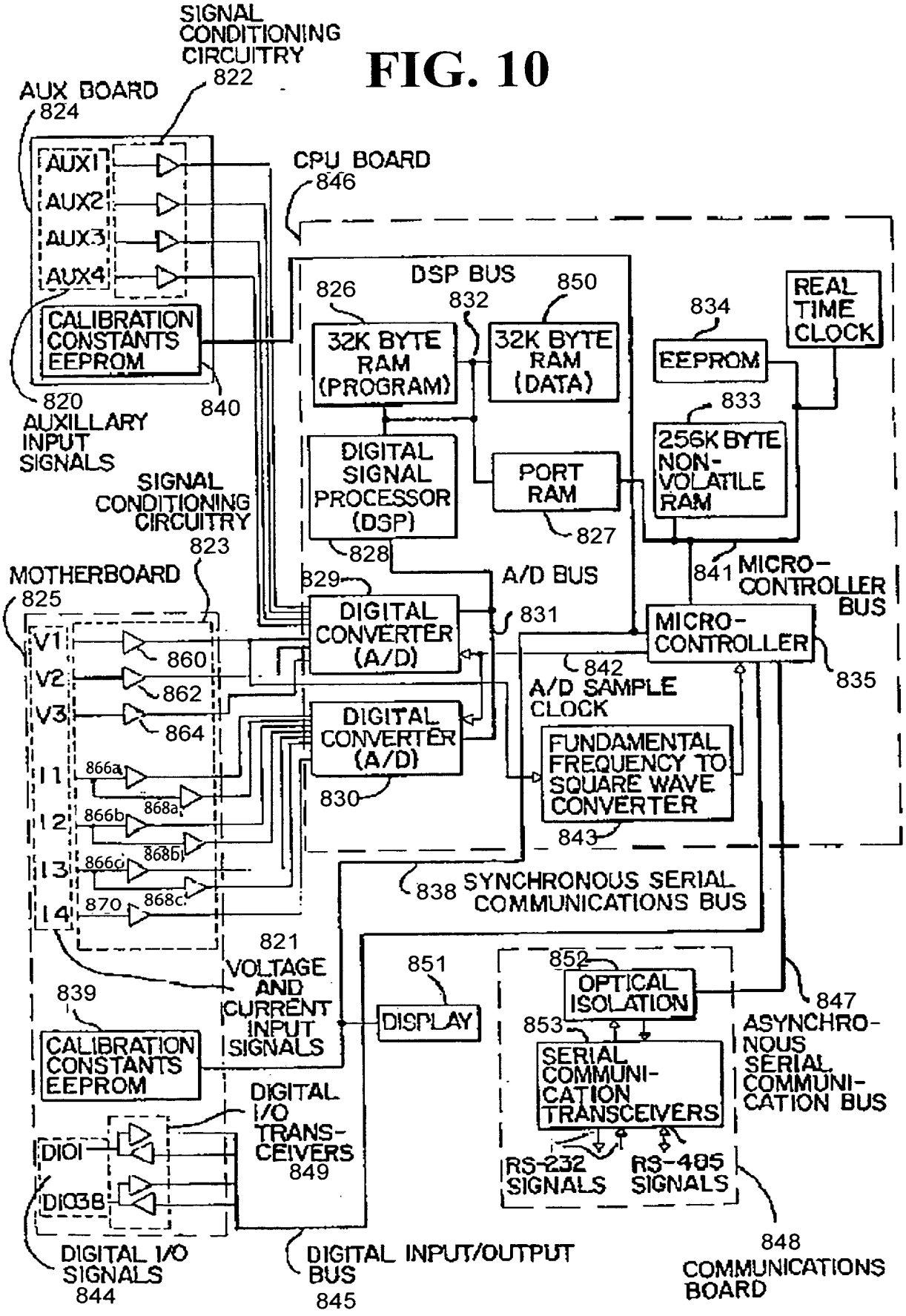


FIG. 10


SIGNAL CONDITIONING CIRCUITRY
822
AUX BOARD
824
AUX1
AUX2
AUX3
AUX4
CALIBRATION CONSTANTS EEPROM
840
820
AUXILLARY INPUT SIGNALS
SIGNAL CONDITIONING CIRCUITRY
823
MOTHERBOARD
825
V1
V2
V3
860
862
864
I1
I2
I3
I4
866a
868a
866b
868b
866c
868c
870
821
VOLTAGE AND CURRENT INPUT SIGNALS
839
CALIBRATION CONSTANTS EEPROM
DIGITAL I/O TRANS-CEIVERS
849
DIO1
DIO38
DIGITAL I/O SIGNALS
844
CPU BOARD
846
DSP BUS
826
832
850
32K BYTE RAM (PROGRAM)
32K BYTE RAM (DATA)
DIGITAL SIGNAL PROCESSOR (DSP)
828
PORT RAM
827
829
A/D BUS
DIGITAL CONVERTER (A/D)
831
DIGITAL CONVERTER (A/D)
830
842
A/D SAMPLE CLOCK
FUNDAMENTAL FREQUENCY TO SQUARE WAVE CONVERTER
843
838
SYNCHRONOUS SERIAL COMMUNICATIONS BUS
DIGITAL INPUT/OUTPUT BUS
845
851
DISPLAY
852
OPTICAL ISOLATION
853
SERIAL COMMUNI-CATION TRANSCEIVERS
RS-232 SIGNALS
RS-485 SIGNALS
847
ASYNCHRO-NOUS SERIAL COMMUNI-CATION BUS
848
COMMUNICATIONS BOARD
834
EEPROM
REAL TIME CLOCK
833
256K BYTE NON-VOLATILE RAM
841
MICRO-CONTROLLER BUS
MICRO-CONTROLLER
835

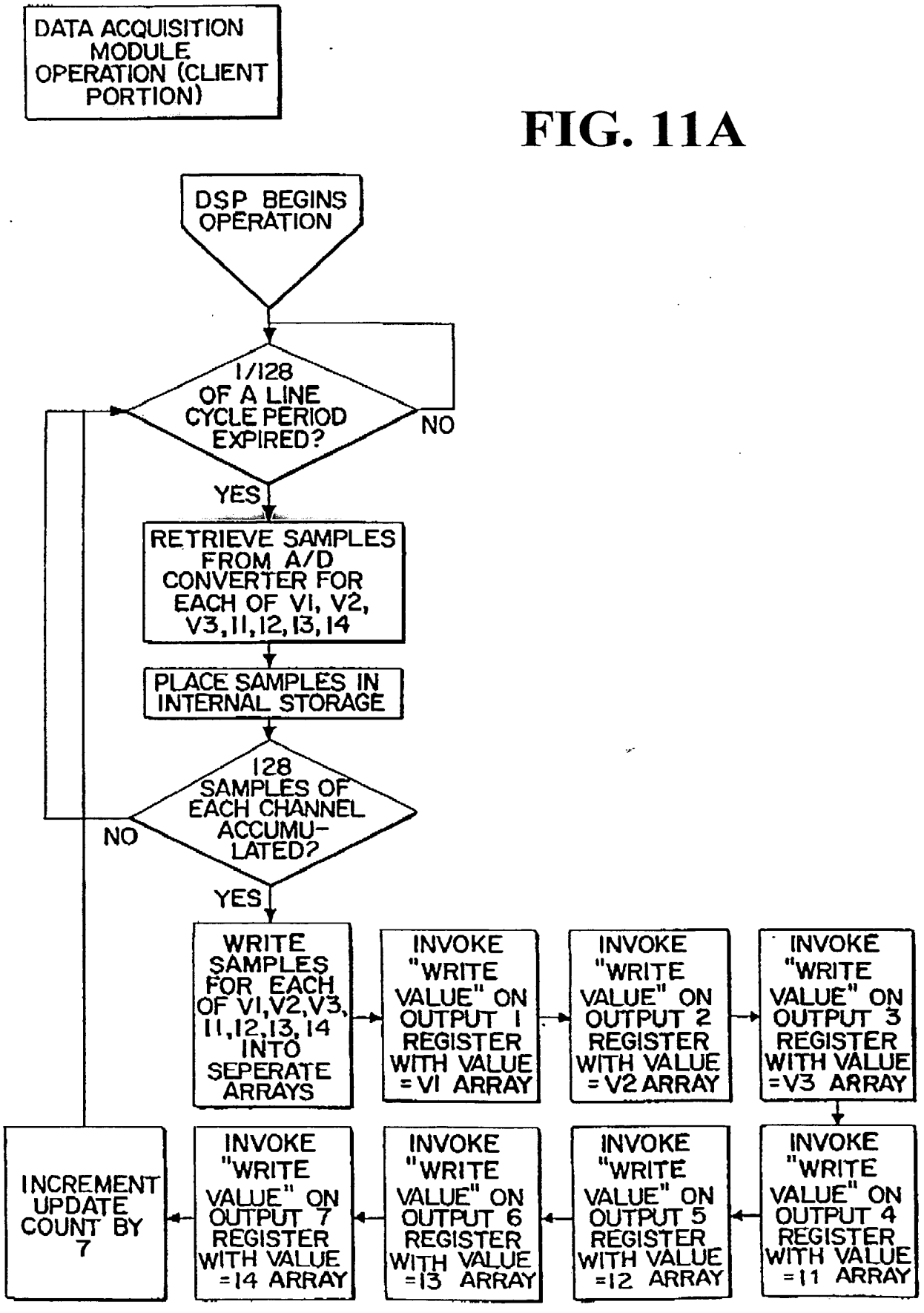
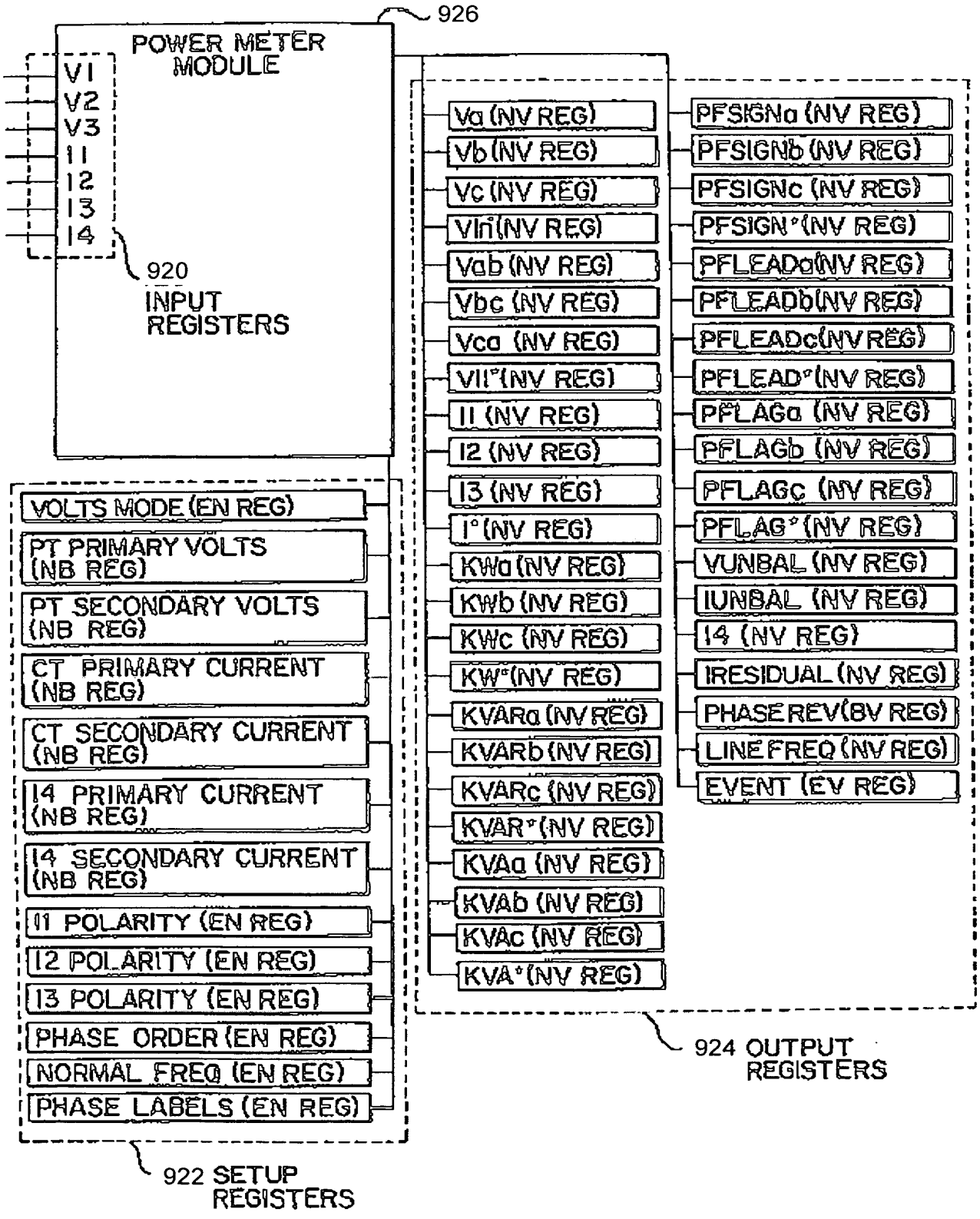

DATA ACQUISITION MODULE OPERATION (CLIENT PORTION)
DSP BEGINS OPERATION
1/128 OF A LINE CYCLE PERIOD EXPIRED?
NO
YES
RETRIEVE SAMPLES FROM A/D CONVERTER FOR EACH OF V1, V2, V3, I1, I2, I3, I4
PLACE SAMPLES IN INTERNAL STORAGE
128 SAMPLES OF EACH CHANNEL ACCUMU-LATED?
NO
YES
WRITE SAMPLES FOR EACH OF V1, V2, V3, I1, I2, I3, I4 INTO SEPERATE ARRAYS
INVOKE "WRITE VALUE" ON OUTPUT 1 REGISTER WITH VALUE =V1 ARRAY
INVOKE "WRITE VALUE" ON OUTPUT 2 REGISTER WITH VALUE =V2 ARRAY
INVOKE "WRITE VALUE" ON OUTPUT 3 REGISTER WITH VALUE =V3 ARRAY
INVOKE "WRITE VALUE" ON OUTPUT 4 REGISTER WITH VALUE =I1 ARRAY
INVOKE "WRITE VALUE" ON OUTPUT 5 REGISTER WITH VALUE =I2 ARRAY
INVOKE "WRITE VALUE" ON OUTPUT 6 REGISTER WITH VALUE =I3 ARRAY
INVOKE "WRITE VALUE" ON OUTPUT 7 REGISTER WITH VALUE =I4 ARRAY
INCREMENT UPDATE COUNT BY 7

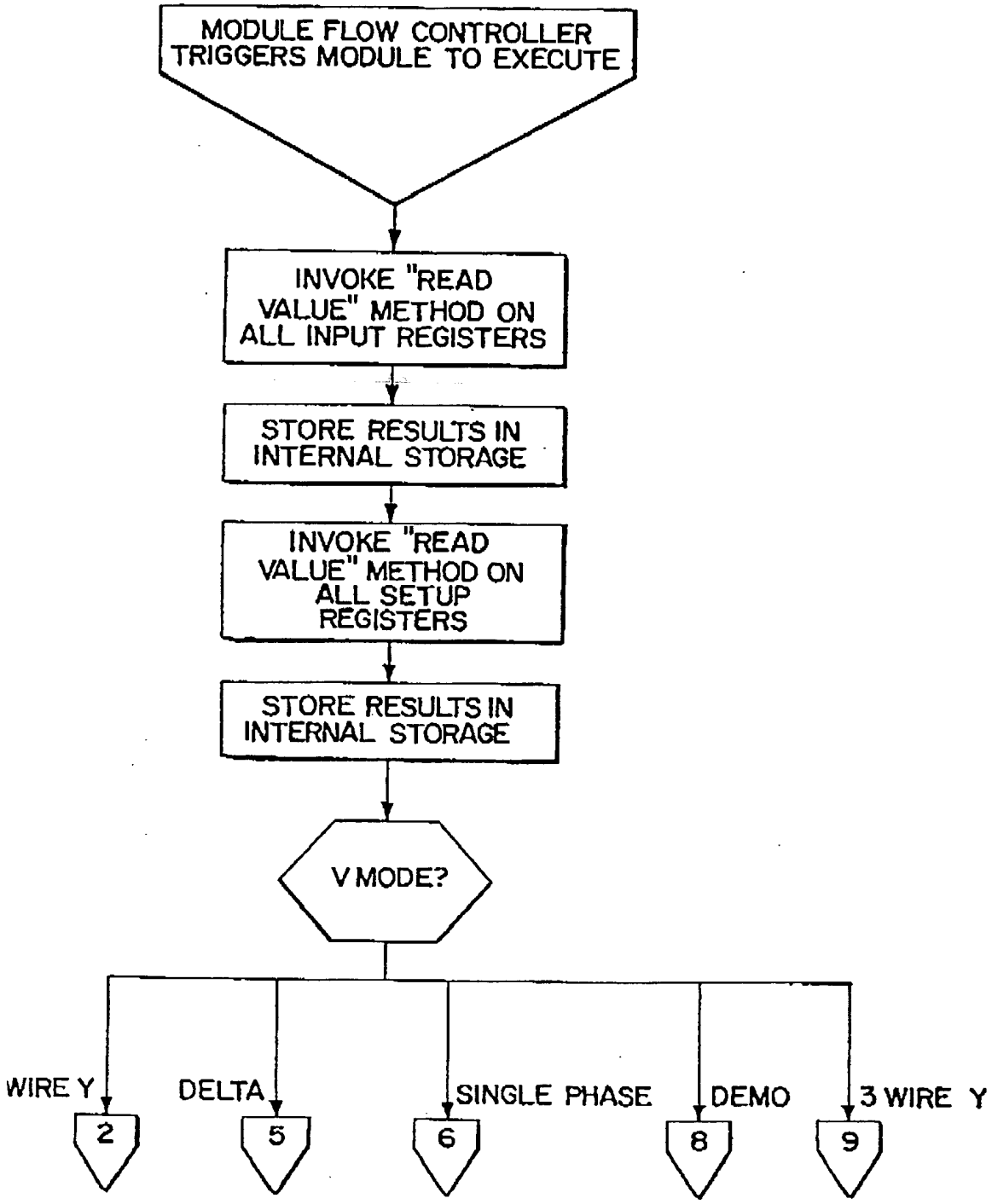
FIG. 11A

FIG. 12


926
POWER METER MODULE
V1
V2
V3
I1
I2
I3
I4
920
INPUT REGISTERS
VOLTS MODE (EN REG)
PT PRIMARY VOLTS (NB REG)
PT SECONDARY VOLTS (NB REG)
CT PRIMARY CURRENT (NB REG)
CT SECONDARY CURRENT (NB REG)
I4 PRIMARY CURRENT (NB REG)
I4 SECONDARY CURRENT (NB REG)
I1 POLARITY (EN REG)
I2 POLARITY (EN REG)
I3 POLARITY (EN REG)
PHASE ORDER (EN REG)
NORMAL FREQ (EN REG)
PHASE LABELS (EN REG)
922 SETUP REGISTERS
Va (NV REG)
Vb (NV REG)
Vc (NV REG)
V1n (NV REG)
Vab (NV REG)
Vbc (NV REG)
Vca (NV REG)
V1° (NV REG)
I1 (NV REG)
I2 (NV REG)
I3 (NV REG)
I° (NV REG)
KWa (NV REG)
KWb (NV REG)
KWc (NV REG)
KW° (NV REG)
KVARa (NV REG)
KVARb (NV REG)
KVARc (NV REG)
KVAR° (NV REG)
KVAa (NV REG)
KVAb (NV REG)
KVAc (NV REG)
KVA° (NV REG)
PFSIGNa (NV REG)
PFSIGNb (NV REG)
PFSIGNc (NV REG)
PFSIGN° (NV REG)
PFLEADa (NV REG)
PFLEADb (NV REG)
PFLEADc (NV REG)
PFLEAD° (NV REG)
PFLAGa (NV REG)
PFLAGb (NV REG)
PFLAGc (NV REG)
PFLAG° (NV REG)
VUNBAL (NV REG)
IUNBAL (NV REG)
I4 (NV REG)
IRESIDUAL (NV REG)
PHASE REV (BV REG)
LINE FREQ (NV REG)
EVENT (EV REG)
924 OUTPUT REGISTERS

POWER METER MODULE OPERATION (CLIENT PORTION)

FIG. 12A



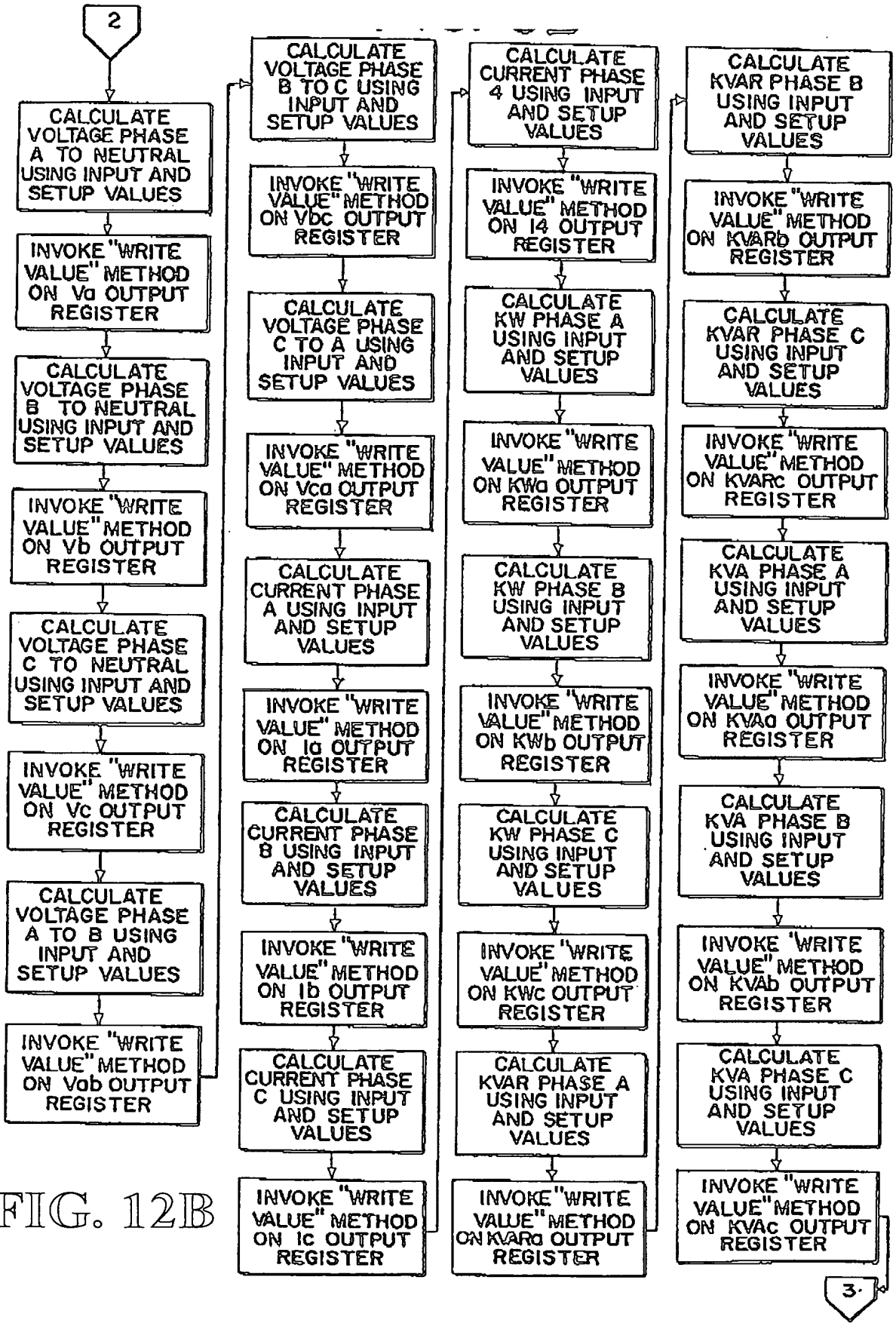

2
CALCULATE VOLTAGE PHASE A TO NEUTRAL USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Va OUTPUT REGISTER
CALCULATE VOLTAGE PHASE B TO NEUTRAL USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Vb OUTPUT REGISTER
CALCULATE VOLTAGE PHASE C TO NEUTRAL USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Vc OUTPUT REGISTER
CALCULATE VOLTAGE PHASE A TO B USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Vab OUTPUT REGISTER
CALCULATE VOLTAGE PHASE B TO C USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Vbc OUTPUT REGISTER
CALCULATE VOLTAGE PHASE C TO A USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Vca OUTPUT REGISTER
CALCULATE CURRENT PHASE A USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Ia OUTPUT REGISTER
CALCULATE CURRENT PHASE B USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Ib OUTPUT REGISTER
CALCULATE CURRENT PHASE C USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Ic OUTPUT REGISTER
CALCULATE CURRENT PHASE 4 USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON I4 OUTPUT REGISTER
CALCULATE KW PHASE A USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KWa OUTPUT REGISTER
CALCULATE KW PHASE B USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KWb OUTPUT REGISTER
CALCULATE KW PHASE C USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KWc OUTPUT REGISTER
CALCULATE KVAR PHASE A USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KVARa OUTPUT REGISTER
CALCULATE KVAR PHASE B USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KVARb OUTPUT REGISTER
CALCULATE KVAR PHASE C USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KVARc OUTPUT REGISTER
CALCULATE KVA PHASE A USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KVAa OUTPUT REGISTER
CALCULATE KVA PHASE B USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KVAb OUTPUT REGISTER
CALCULATE KVA PHASE C USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KVAc OUTPUT REGISTER
3

FIG. 12B

FIG. 12C

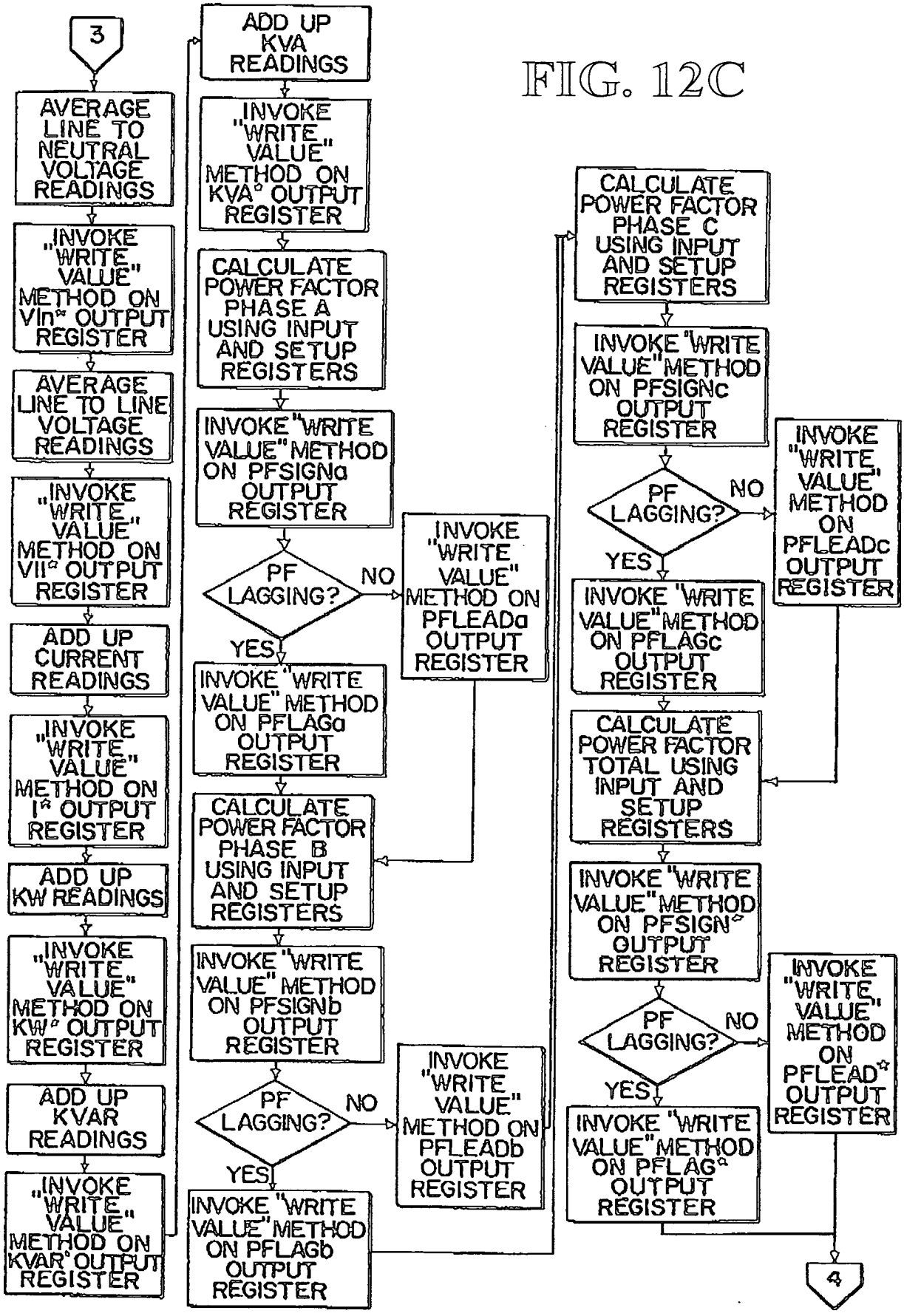

3
AVERAGE LINE TO NEUTRAL VOLTAGE READINGS
INVOKE "WRITE VALUE" METHOD ON Vln* OUTPUT REGISTER
AVERAGE LINE TO LINE VOLTAGE READINGS
INVOKE "WRITE VALUE" METHOD ON Vll* OUTPUT REGISTER
ADD UP CURRENT READINGS
INVOKE "WRITE VALUE" METHOD ON I* OUTPUT REGISTER
ADD UP KW READINGS
INVOKE "WRITE VALUE" METHOD ON KW* OUTPUT REGISTER
ADD UP KVAR READINGS
INVOKE "WRITE VALUE" METHOD ON KVAR* OUTPUT REGISTER
ADD UP KVA READINGS
INVOKE "WRITE VALUE" METHOD ON KVA* OUTPUT REGISTER
CALCULATE POWER FACTOR PHASE A USING INPUT AND SETUP REGISTERS
INVOKE "WRITE VALUE" METHOD ON PFSIGNa OUTPUT REGISTER
PF LAGGING?
NO
YES
INVOKE "WRITE VALUE" METHOD ON PFLEADa OUTPUT REGISTER
INVOKE "WRITE VALUE" METHOD ON PFLAGa OUTPUT REGISTER
CALCULATE POWER FACTOR PHASE B USING INPUT AND SETUP REGISTERS
INVOKE "WRITE VALUE" METHOD ON PFSIGNb OUTPUT REGISTER
PF LAGGING?
NO
YES
INVOKE "WRITE VALUE" METHOD ON PFLEADb OUTPUT REGISTER
INVOKE "WRITE VALUE" METHOD ON PFLAGb OUTPUT REGISTER
CALCULATE POWER FACTOR PHASE C USING INPUT AND SETUP REGISTERS
INVOKE "WRITE VALUE" METHOD ON PFSIGNc OUTPUT REGISTER
PF LAGGING?
NO
YES
INVOKE "WRITE VALUE" METHOD ON PFLEADc OUTPUT REGISTER
INVOKE "WRITE VALUE" METHOD ON PFLAGc OUTPUT REGISTER
CALCULATE POWER FACTOR TOTAL USING INPUT AND SETUP REGISTERS
INVOKE "WRITE VALUE" METHOD ON PFSIGN* OUTPUT REGISTER
PF LAGGING?
NO
YES
INVOKE "WRITE VALUE" METHOD ON PFLEAD* OUTPUT REGISTER
INVOKE "WRITE VALUE" METHOD ON PFLAG* OUTPUT REGISTER
4

FIG. 12D

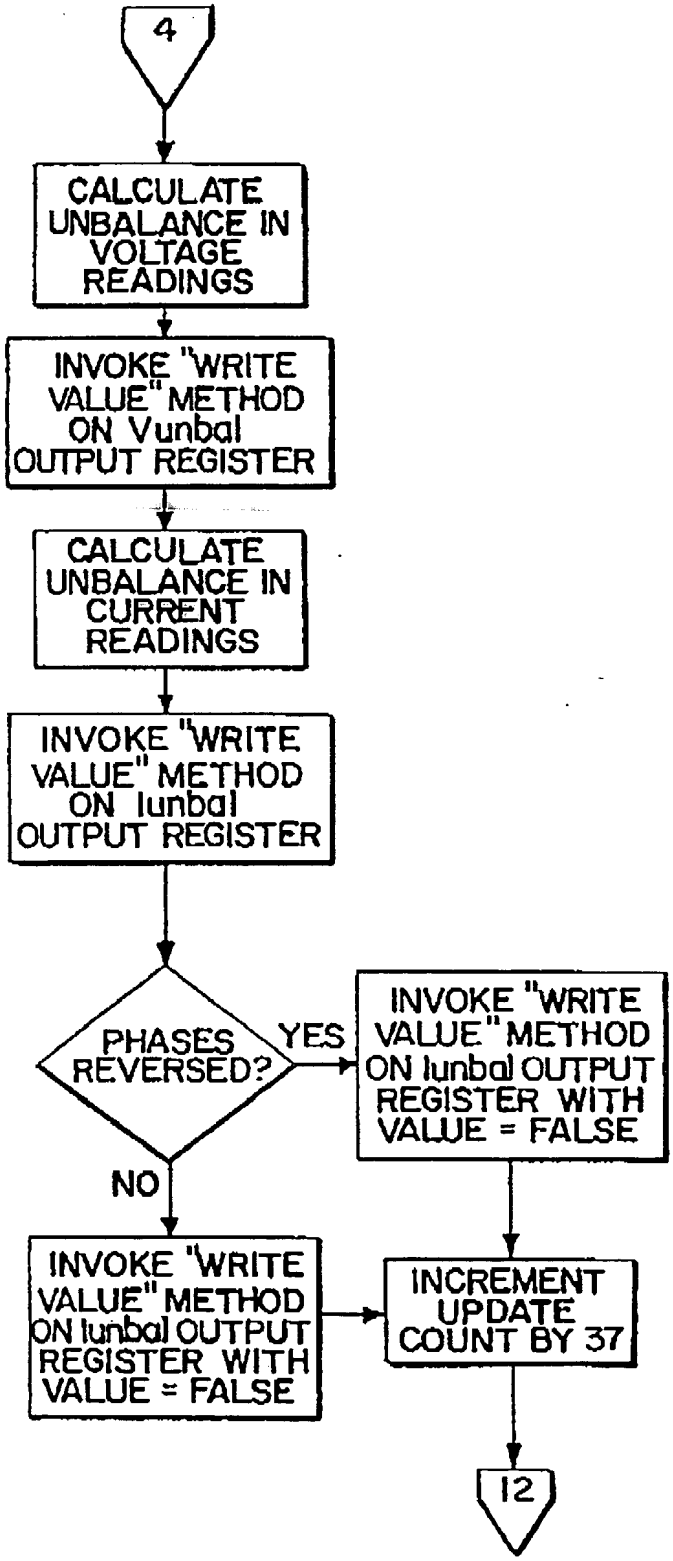


FIG. 12H

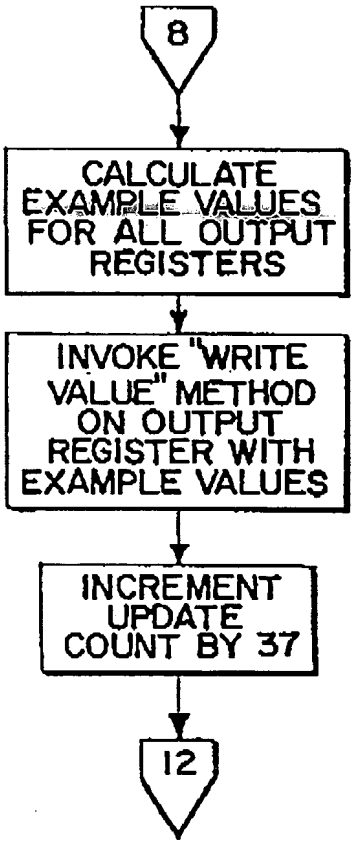


FIG. 12E

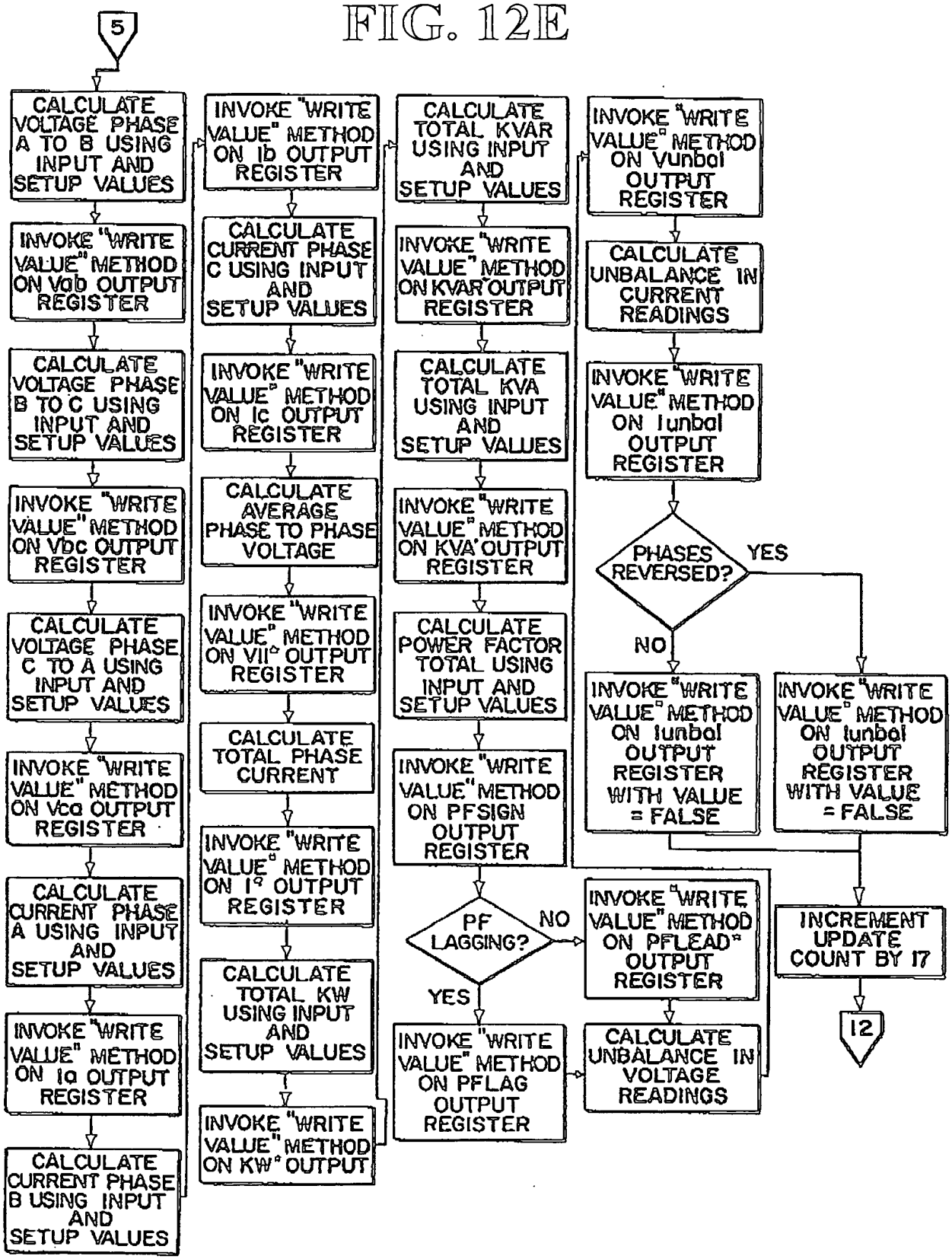

5
CALCULATE VOLTAGE PHASE A TO B USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Vab OUTPUT REGISTER
CALCULATE VOLTAGE PHASE B TO C USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Vbc OUTPUT REGISTER
CALCULATE VOLTAGE PHASE C TO A USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Vca OUTPUT REGISTER
CALCULATE CURRENT PHASE A USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Ia OUTPUT REGISTER
CALCULATE CURRENT PHASE B USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Ib OUTPUT REGISTER
CALCULATE CURRENT PHASE C USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Ic OUTPUT REGISTER
CALCULATE AVERAGE PHASE TO PHASE VOLTAGE
INVOKE "WRITE VALUE" METHOD ON Vll° OUTPUT REGISTER
CALCULATE TOTAL PHASE CURRENT
INVOKE "WRITE VALUE" METHOD ON I° OUTPUT REGISTER
CALCULATE TOTAL KW USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KW° OUTPUT
CALCULATE TOTAL KVAR USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KVAR° OUTPUT REGISTER
CALCULATE TOTAL KVA USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KVA° OUTPUT REGISTER
CALCULATE POWER FACTOR TOTAL USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON PFSIGN OUTPUT REGISTER
PF LAGGING?
NO
YES
INVOKE "WRITE VALUE" METHOD ON PFLEAD° OUTPUT REGISTER
INVOKE "WRITE VALUE" METHOD ON PFLAG OUTPUT REGISTER
INVOKE "WRITE VALUE" METHOD ON Vunbal OUTPUT REGISTER
CALCULATE UNBALANCE IN CURRENT READINGS
INVOKE "WRITE VALUE" METHOD ON Iunbal OUTPUT REGISTER
PHASES REVERSED?
YES
NO
INVOKE "WRITE VALUE" METHOD ON Iunbal OUTPUT REGISTER WITH VALUE = FALSE
INVOKE "WRITE VALUE" METHOD ON Iunbal OUTPUT REGISTER WITH VALUE = FALSE
CALCULATE UNBALANCE IN VOLTAGE READINGS
INCREMENT UPDATE COUNT BY 17
12

FIG. 12F

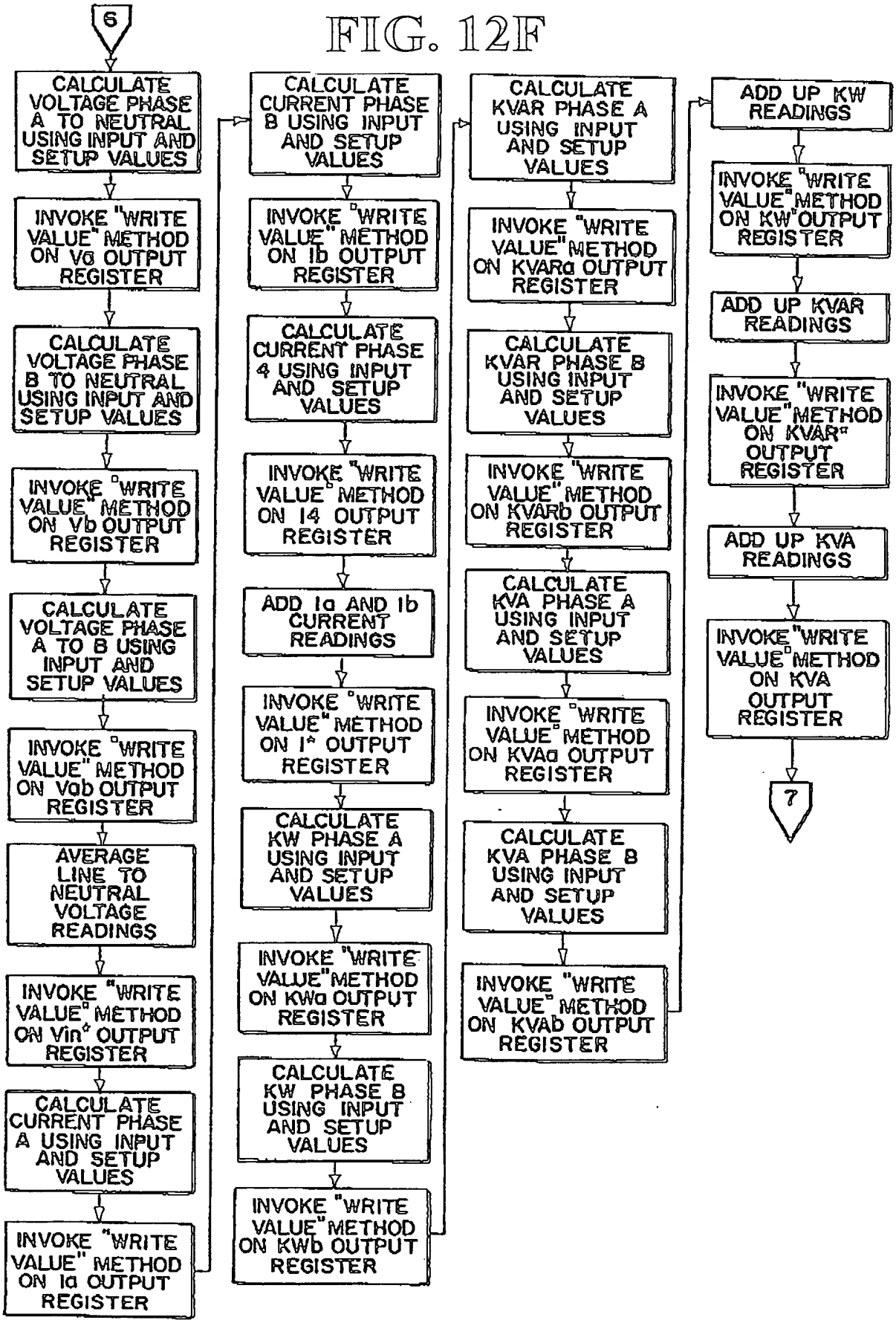
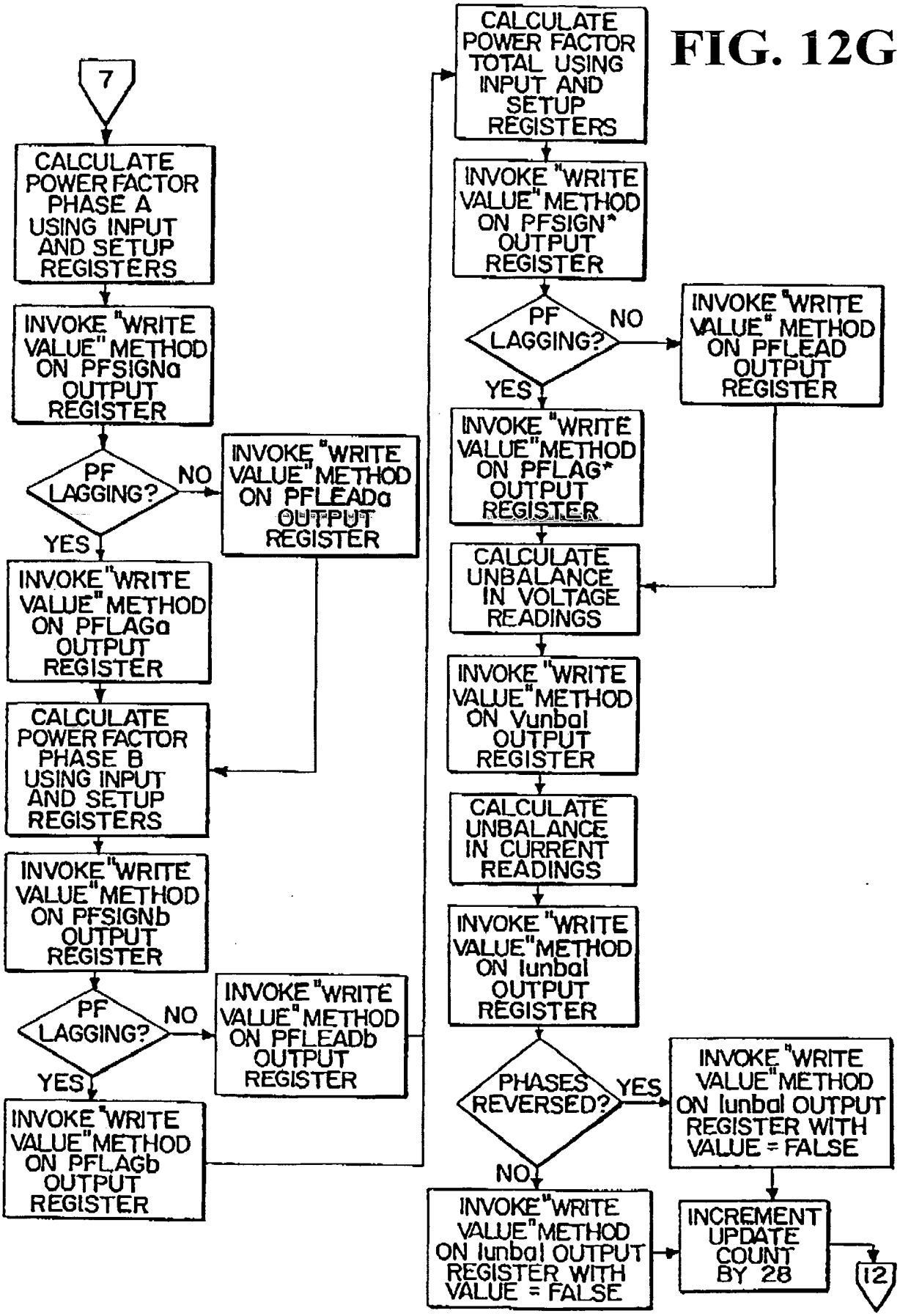

6
CALCULATE VOLTAGE PHASE A TO NEUTRAL USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Va OUTPUT REGISTER
CALCULATE VOLTAGE PHASE B TO NEUTRAL USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Vb OUTPUT REGISTER
CALCULATE VOLTAGE PHASE A TO B USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Vab OUTPUT REGISTER
AVERAGE LINE TO NEUTRAL VOLTAGE READINGS
INVOKE "WRITE VALUE" METHOD ON Vin* OUTPUT REGISTER
CALCULATE CURRENT PHASE A USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Ia OUTPUT REGISTER
CALCULATE CURRENT PHASE B USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Ib OUTPUT REGISTER
CALCULATE CURRENT PHASE 4 USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON I4 OUTPUT REGISTER
ADD Ia AND Ib CURRENT READINGS
INVOKE "WRITE VALUE" METHOD ON I* OUTPUT REGISTER
CALCULATE KW PHASE A USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KWa OUTPUT REGISTER
CALCULATE KW PHASE B USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KWb OUTPUT REGISTER
CALCULATE KVAR PHASE A USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KVARa OUTPUT REGISTER
CALCULATE KVAR PHASE B USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KVARb OUTPUT REGISTER
CALCULATE KVA PHASE A USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KVAa OUTPUT REGISTER
CALCULATE KVA PHASE B USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KVAb OUTPUT REGISTER
ADD UP KW READINGS
INVOKE "WRITE VALUE" METHOD ON KW" OUTPUT REGISTER
ADD UP KVAR READINGS
INVOKE "WRITE VALUE" METHOD ON KVAR" OUTPUT REGISTER
ADD UP KVA READINGS
INVOKE "WRITE VALUE" METHOD ON KVA OUTPUT REGISTER
7

FIG. 12G


7
CALCULATE POWER FACTOR PHASE A USING INPUT AND SETUP REGISTERS
INVOKE "WRITE VALUE" METHOD ON PFSIGNa OUTPUT REGISTER
PF LAGGING?
NO
YES
INVOKE "WRITE VALUE" METHOD ON PFLEADa OUTPUT REGISTER
INVOKE "WRITE VALUE" METHOD ON PFLAGa OUTPUT REGISTER
CALCULATE POWER FACTOR PHASE B USING INPUT AND SETUP REGISTERS
INVOKE "WRITE VALUE" METHOD ON PFSIGNb OUTPUT REGISTER
PF LAGGING?
NO
YES
INVOKE "WRITE VALUE" METHOD ON PFLEADb OUTPUT REGISTER
INVOKE "WRITE VALUE" METHOD ON PFLAGb OUTPUT REGISTER
CALCULATE POWER FACTOR TOTAL USING INPUT AND SETUP REGISTERS
INVOKE "WRITE VALUE" METHOD ON PFSIGN* OUTPUT REGISTER
PF LAGGING?
NO
YES
INVOKE "WRITE VALUE" METHOD ON PFLEAD OUTPUT REGISTER
INVOKE "WRITE VALUE" METHOD ON PFLAG* OUTPUT REGISTER
CALCULATE UNBALANCE IN VOLTAGE READINGS
INVOKE "WRITE VALUE" METHOD ON Vunbal OUTPUT REGISTER
CALCULATE UNBALANCE IN CURRENT READINGS
INVOKE "WRITE VALUE" METHOD ON Iunbal OUTPUT REGISTER
PHASES REVERSED?
YES
NO
INVOKE "WRITE VALUE" METHOD ON Iunbal OUTPUT REGISTER WITH VALUE = FALSE
INVOKE "WRITE VALUE" METHOD ON Iunbal OUTPUT REGISTER WITH VALUE = FALSE
INCREMENT UPDATE COUNT BY 28
12

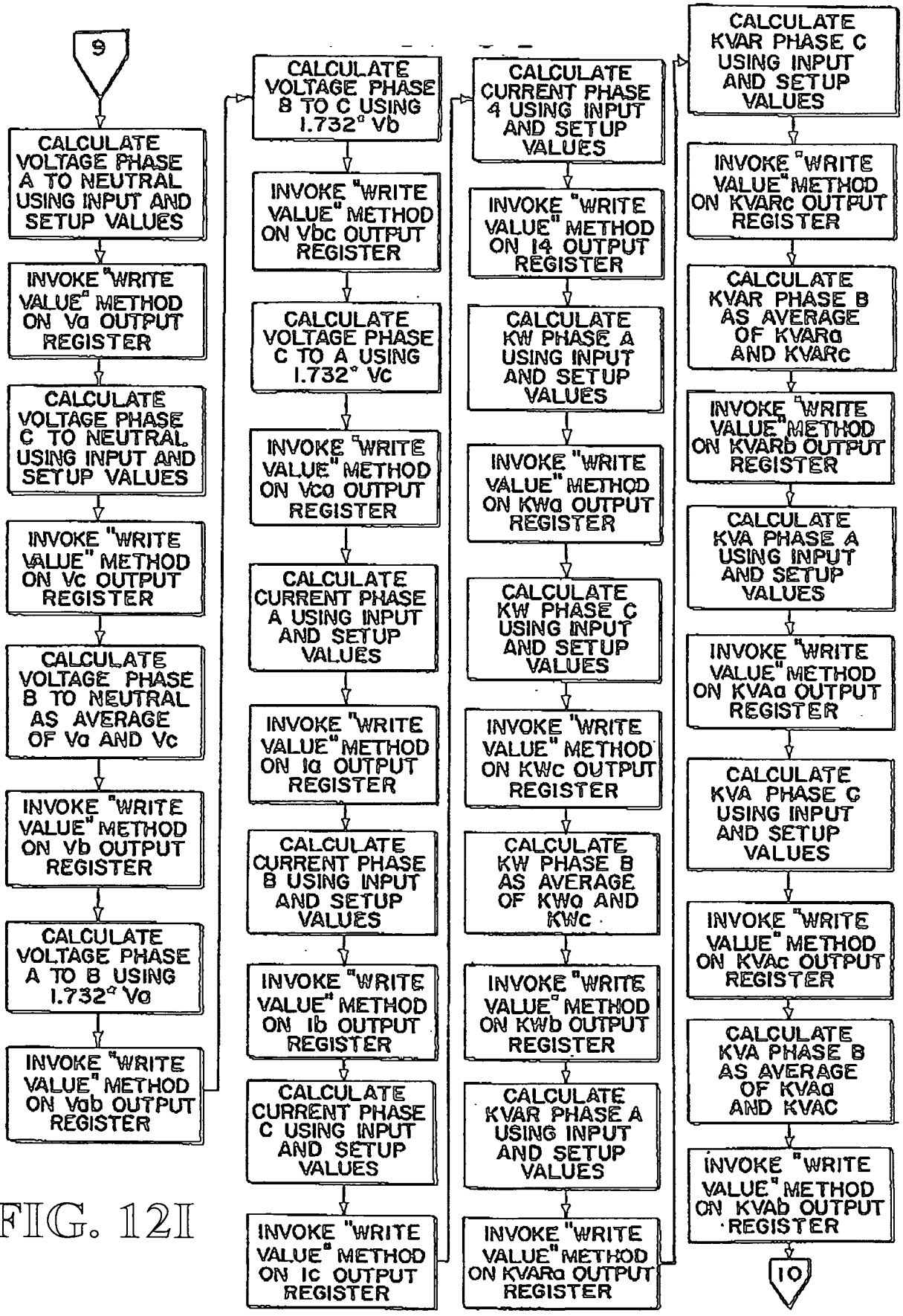

9
CALCULATE VOLTAGE PHASE A TO NEUTRAL USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Va OUTPUT REGISTER
CALCULATE VOLTAGE PHASE C TO NEUTRAL USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Vc OUTPUT REGISTER
CALCULATE VOLTAGE PHASE B TO NEUTRAL AS AVERAGE OF Va AND Vc
INVOKE "WRITE VALUE" METHOD ON Vb OUTPUT REGISTER
CALCULATE VOLTAGE PHASE A TO B USING 1.732° Va
INVOKE "WRITE VALUE" METHOD ON Vab OUTPUT REGISTER
CALCULATE VOLTAGE PHASE B TO C USING 1.732° Vb
INVOKE "WRITE VALUE" METHOD ON Vbc OUTPUT REGISTER
CALCULATE VOLTAGE PHASE C TO A USING 1.732° Vc
INVOKE "WRITE VALUE" METHOD ON Vca OUTPUT REGISTER
CALCULATE CURRENT PHASE A USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Ia OUTPUT REGISTER
CALCULATE CURRENT PHASE B USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Ib OUTPUT REGISTER
CALCULATE CURRENT PHASE C USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON Ic OUTPUT REGISTER
CALCULATE CURRENT PHASE 4 USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON I4 OUTPUT REGISTER
CALCULATE KW PHASE A USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KWa OUTPUT REGISTER
CALCULATE KW PHASE C USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KWc OUTPUT REGISTER
CALCULATE KW PHASE B AS AVERAGE OF KWa AND KWc
INVOKE "WRITE VALUE" METHOD ON KWb OUTPUT REGISTER
CALCULATE KVAR PHASE A USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KVARa OUTPUT REGISTER
CALCULATE KVAR PHASE C USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KVARc OUTPUT REGISTER
CALCULATE KVAR PHASE B AS AVERAGE OF KVARa AND KVARc
INVOKE "WRITE VALUE" METHOD ON KVARb OUTPUT REGISTER
CALCULATE KVA PHASE A USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KVAa OUTPUT REGISTER
CALCULATE KVA PHASE C USING INPUT AND SETUP VALUES
INVOKE "WRITE VALUE" METHOD ON KVAc OUTPUT REGISTER
CALCULATE KVA PHASE B AS AVERAGE OF KVAa AND KVAC
INVOKE "WRITE VALUE" METHOD ON KVAb OUTPUT REGISTER
10

FIG. 12I

FIG. 12J

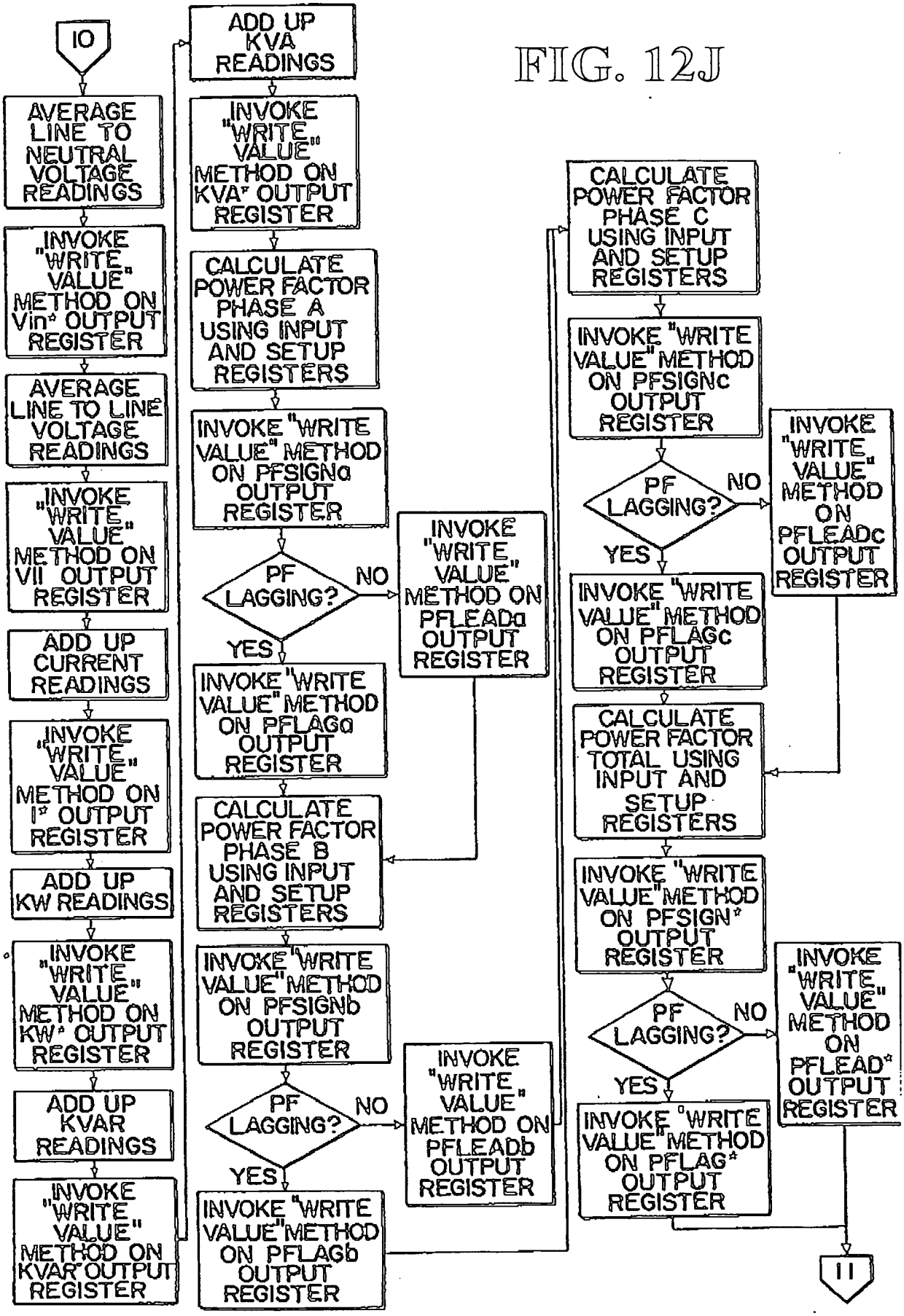

10
AVERAGE LINE TO NEUTRAL VOLTAGE READINGS
INVOKE "WRITE VALUE" METHOD ON Vin* OUTPUT REGISTER
AVERAGE LINE TO LINE VOLTAGE READINGS
INVOKE "WRITE VALUE" METHOD ON VII OUTPUT REGISTER
ADD UP CURRENT READINGS
INVOKE "WRITE VALUE" METHOD ON I* OUTPUT REGISTER
ADD UP KW READINGS
INVOKE "WRITE VALUE" METHOD ON KW* OUTPUT REGISTER
ADD UP KVAR READINGS
INVOKE "WRITE VALUE" METHOD ON KVAR OUTPUT REGISTER
ADD UP KVA READINGS
INVOKE "WRITE VALUE" METHOD ON KVA* OUTPUT REGISTER
CALCULATE POWER FACTOR PHASE A USING INPUT AND SETUP REGISTERS
INVOKE "WRITE VALUE" METHOD ON PFSIGNa OUTPUT REGISTER
PF LAGGING?
NO
YES
INVOKE "WRITE VALUE" METHOD ON PFLEADa OUTPUT REGISTER
INVOKE "WRITE VALUE" METHOD ON PFLAGa OUTPUT REGISTER
CALCULATE POWER FACTOR PHASE B USING INPUT AND SETUP REGISTERS
INVOKE "WRITE VALUE" METHOD ON PFSIGNb OUTPUT REGISTER
PF LAGGING?
NO
YES
INVOKE "WRITE VALUE" METHOD ON PFLEADb OUTPUT REGISTER
INVOKE "WRITE VALUE" METHOD ON PFLAGb OUTPUT REGISTER
CALCULATE POWER FACTOR PHASE C USING INPUT AND SETUP REGISTERS
INVOKE "WRITE VALUE" METHOD ON PFSIGNc OUTPUT REGISTER
PF LAGGING?
NO
YES
INVOKE "WRITE VALUE" METHOD ON PFLEADc OUTPUT REGISTER
INVOKE "WRITE VALUE" METHOD ON PFLAGc OUTPUT REGISTER
CALCULATE POWER FACTOR TOTAL USING INPUT AND SETUP REGISTERS
INVOKE "WRITE VALUE" METHOD ON PFSIGN* OUTPUT REGISTER
PF LAGGING?
NO
YES
INVOKE "WRITE VALUE" METHOD ON PFLEAD* OUTPUT REGISTER
INVOKE "WRITE VALUE" METHOD ON PFLAG* OUTPUT REGISTER
11

FIG. 12K

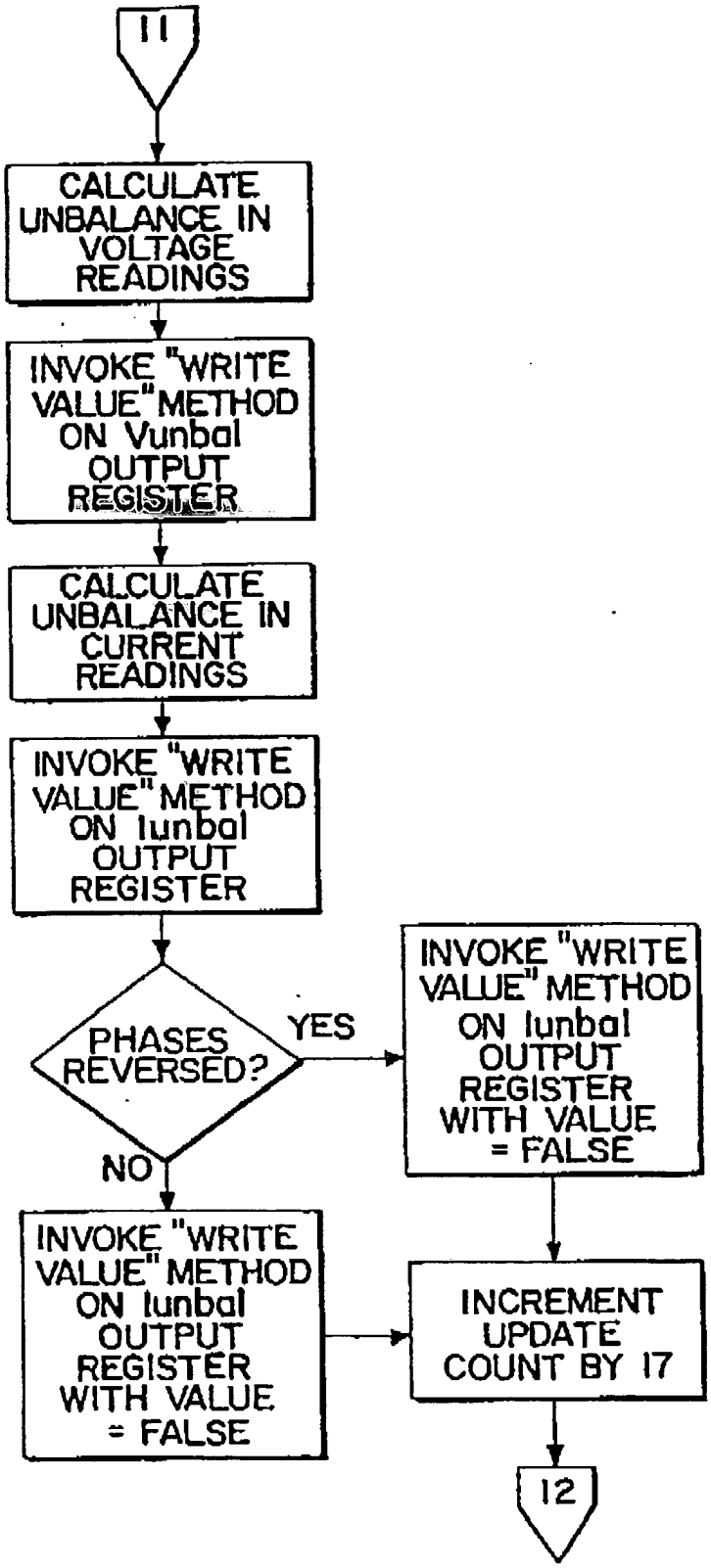


FIG. 12L

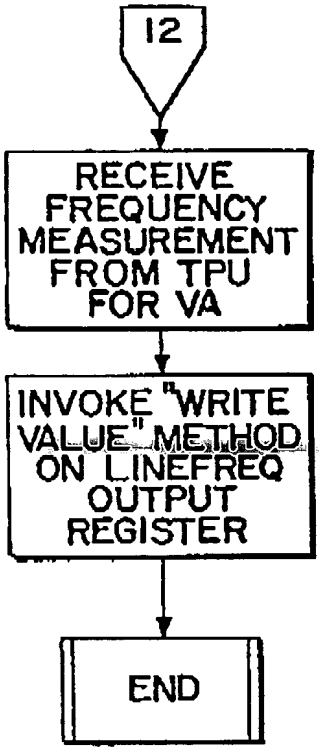


FIG. 13

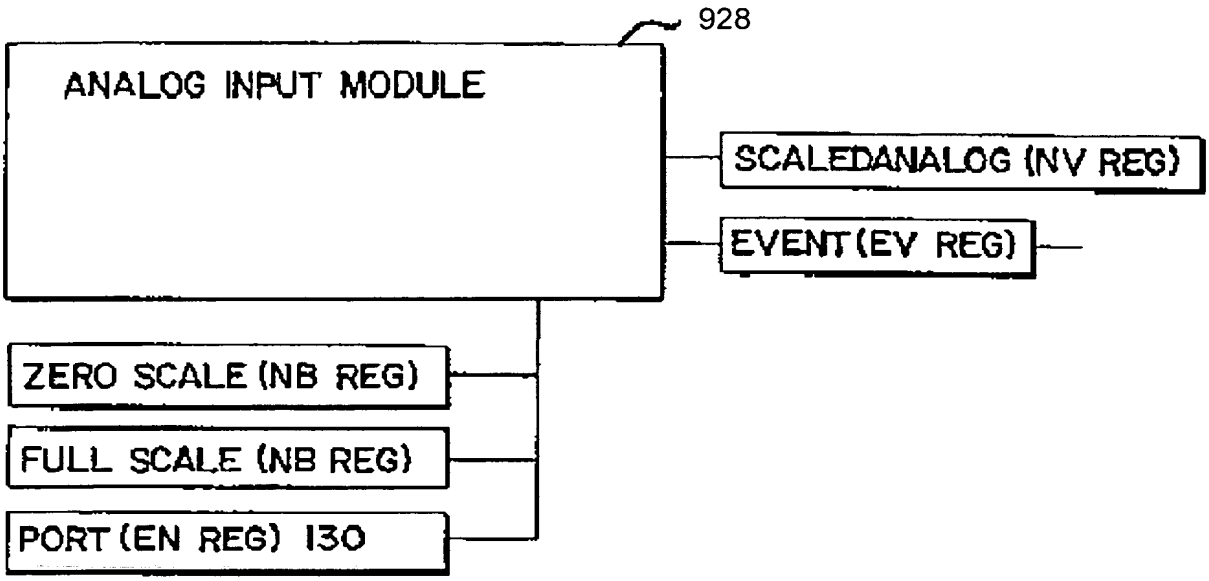

928
ANALOG INPUT MODULE
SCALEDANALOG (NV REG)
EVENT (EV REG)
ZERO SCALE (NB REG)
FULL SCALE (NB REG)
PORT (EN REG) 130

FIG. 14

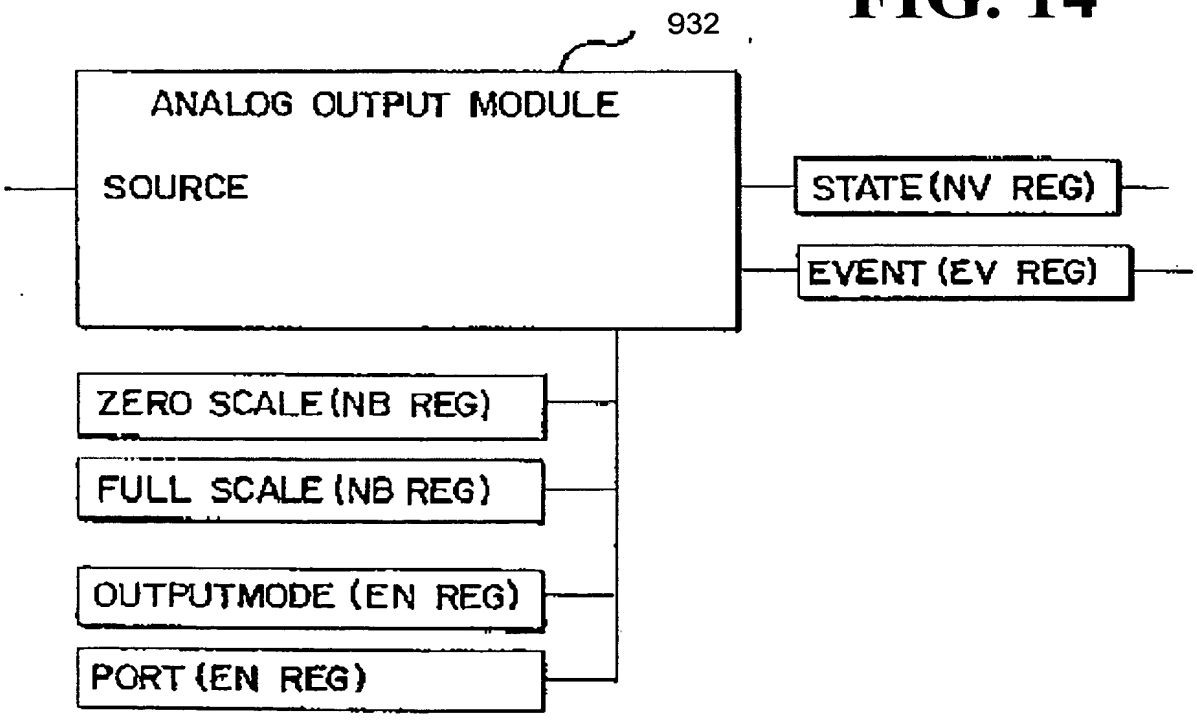

932
ANALOG OUTPUT MODULE
SOURCE
STATE (NV REG)
EVENT (EV REG)
ZERO SCALE (NB REG)
FULL SCALE (NB REG)
OUTPUTMODE (EN REG)
PORT (EN REG)

FIG. 13A

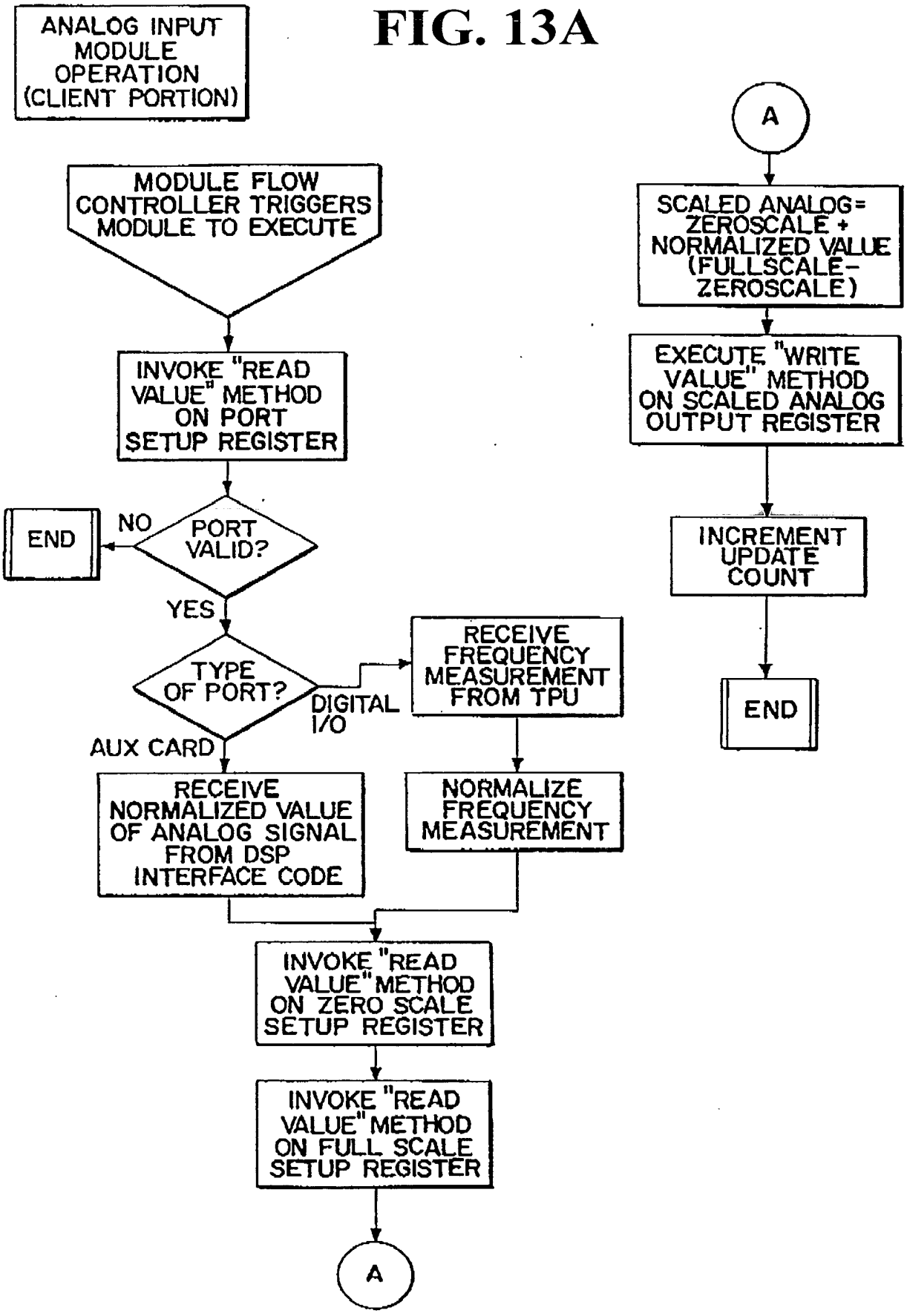

ANALOG INPUT MODULE OPERATION (CLIENT PORTION)
MODULE FLOW CONTROLLER TRIGGERS MODULE TO EXECUTE
INVOKE "READ VALUE" METHOD ON PORT SETUP REGISTER
PORT VALID?
NO
END
YES
TYPE OF PORT?
DIGITAL I/O
AUX CARD
RECEIVE FREQUENCY MEASUREMENT FROM TPU
RECEIVE NORMALIZED VALUE OF ANALOG SIGNAL FROM DSP INTERFACE CODE
NORMALIZE FREQUENCY MEASUREMENT
INVOKE "READ VALUE" METHOD ON ZERO SCALE SETUP REGISTER
INVOKE "READ VALUE" METHOD ON FULL SCALE SETUP REGISTER
A
A
SCALED ANALOG= ZEROSCALE + NORMALIZED VALUE (FULLSCALE- ZEROSCALE)
EXECUTE "WRITE VALUE" METHOD ON SCALED ANALOG OUTPUT REGISTER
INCREMENT UPDATE COUNT
END

FIG. 14A

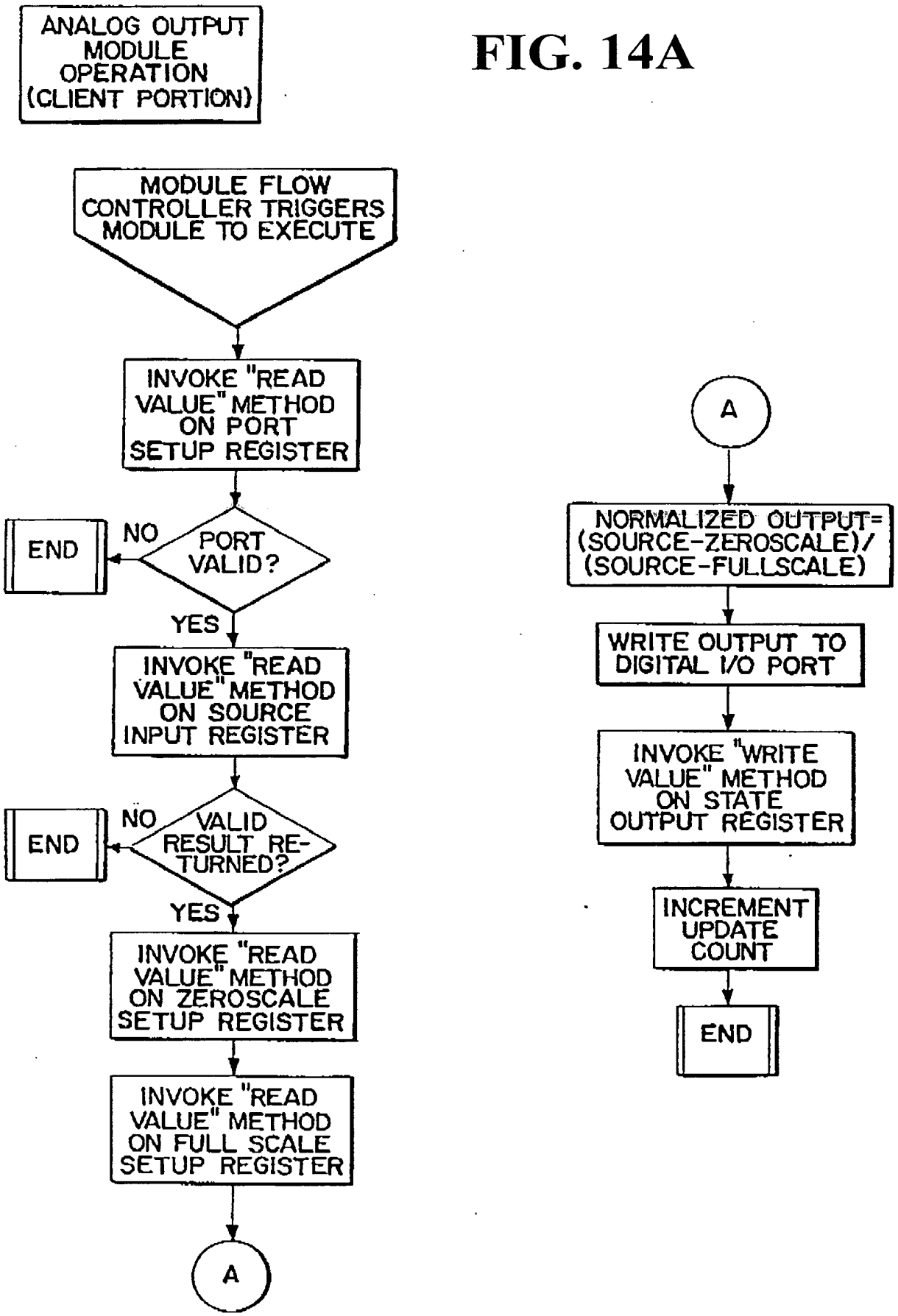

ANALOG OUTPUT MODULE OPERATION (CLIENT PORTION)
MODULE FLOW CONTROLLER TRIGGERS MODULE TO EXECUTE
INVOKE "READ VALUE" METHOD ON PORT SETUP REGISTER
PORT VALID?
NO
END
YES
INVOKE "READ VALUE" METHOD ON SOURCE INPUT REGISTER
VALID RESULT RE-TURNED?
NO
END
YES
INVOKE "READ VALUE" METHOD ON ZEROSCALE SETUP REGISTER
INVOKE "READ VALUE" METHOD ON FULL SCALE SETUP REGISTER
A
A
NORMALIZED OUTPUT= (SOURCE-ZEROSCALE)/ (SOURCE-FULLSCALE)
WRITE OUTPUT TO DIGITAL I/O PORT
INVOKE "WRITE VALUE" METHOD ON STATE OUTPUT REGISTER
INCREMENT UPDATE COUNT
END

FIG. 15

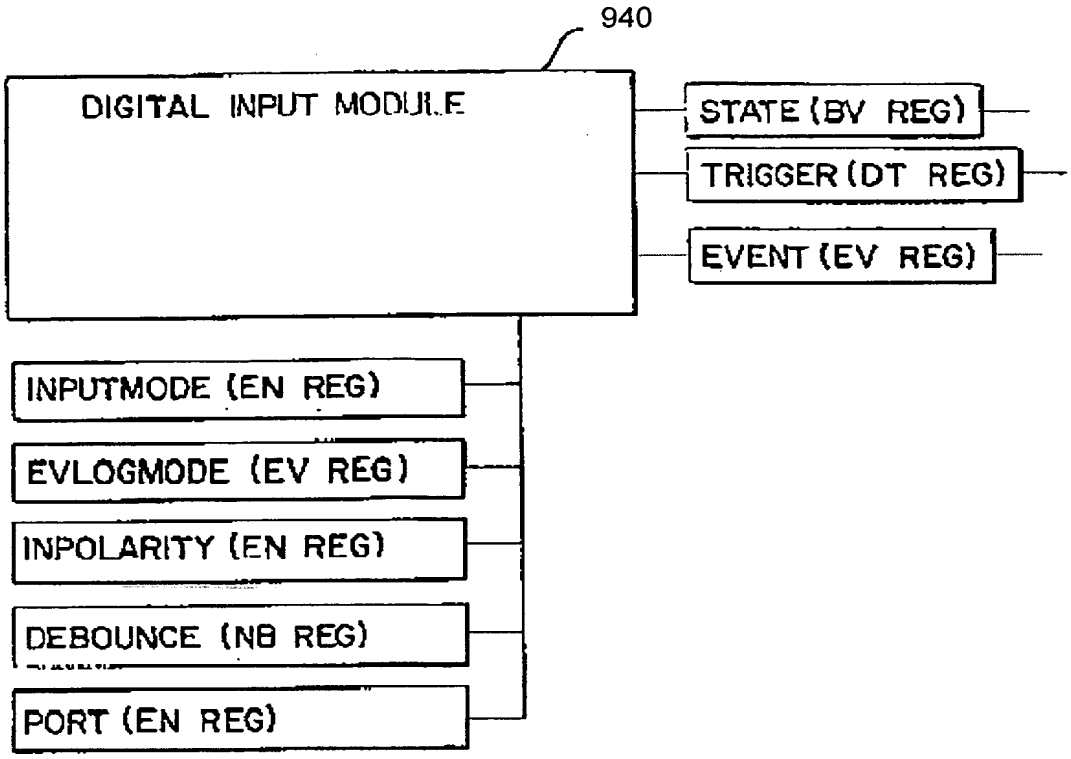

940
DIGITAL INPUT MODULE
STATE (BV REG)
TRIGGER (DT REG)
EVENT (EV REG)
INPUTMODE (EN REG)
EVLOGMODE (EV REG)
INPOLARITY (EN REG)
DEBOUNCE (NB REG)
PORT (EN REG)

FIG. 16

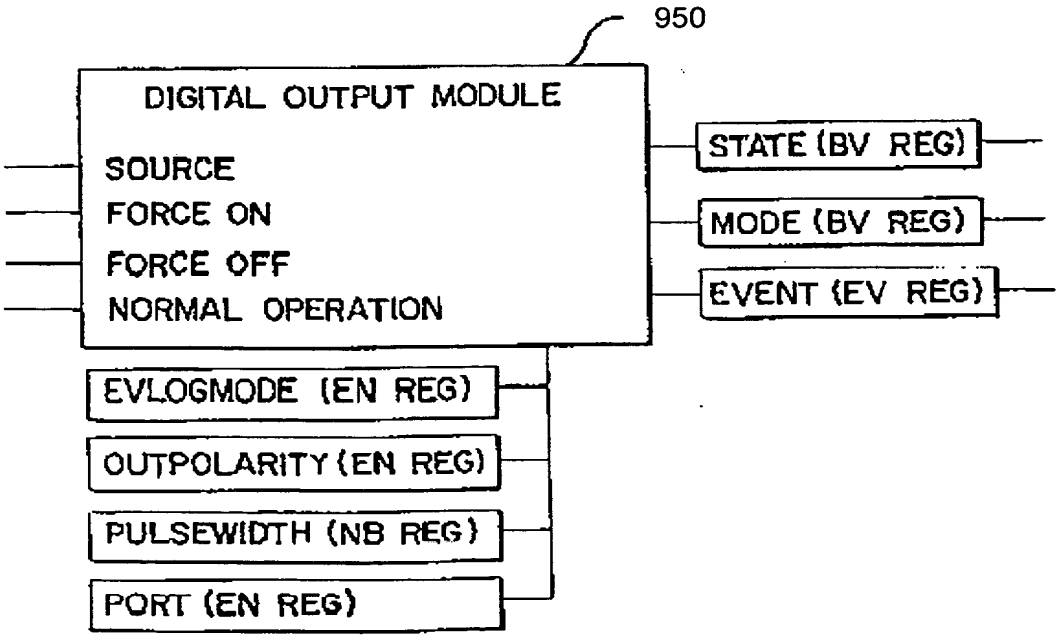

950
DIGITAL OUTPUT MODULE
SOURCE
FORCE ON
FORCE OFF
NORMAL OPERATION
STATE (BV REG)
MODE (BV REG)
EVENT (EV REG)
EVLOGMODE (EN REG)
OUTPOLARITY (EN REG)
PULSEWIDTH (NB REG)
PORT (EN REG)

FIG. 15A

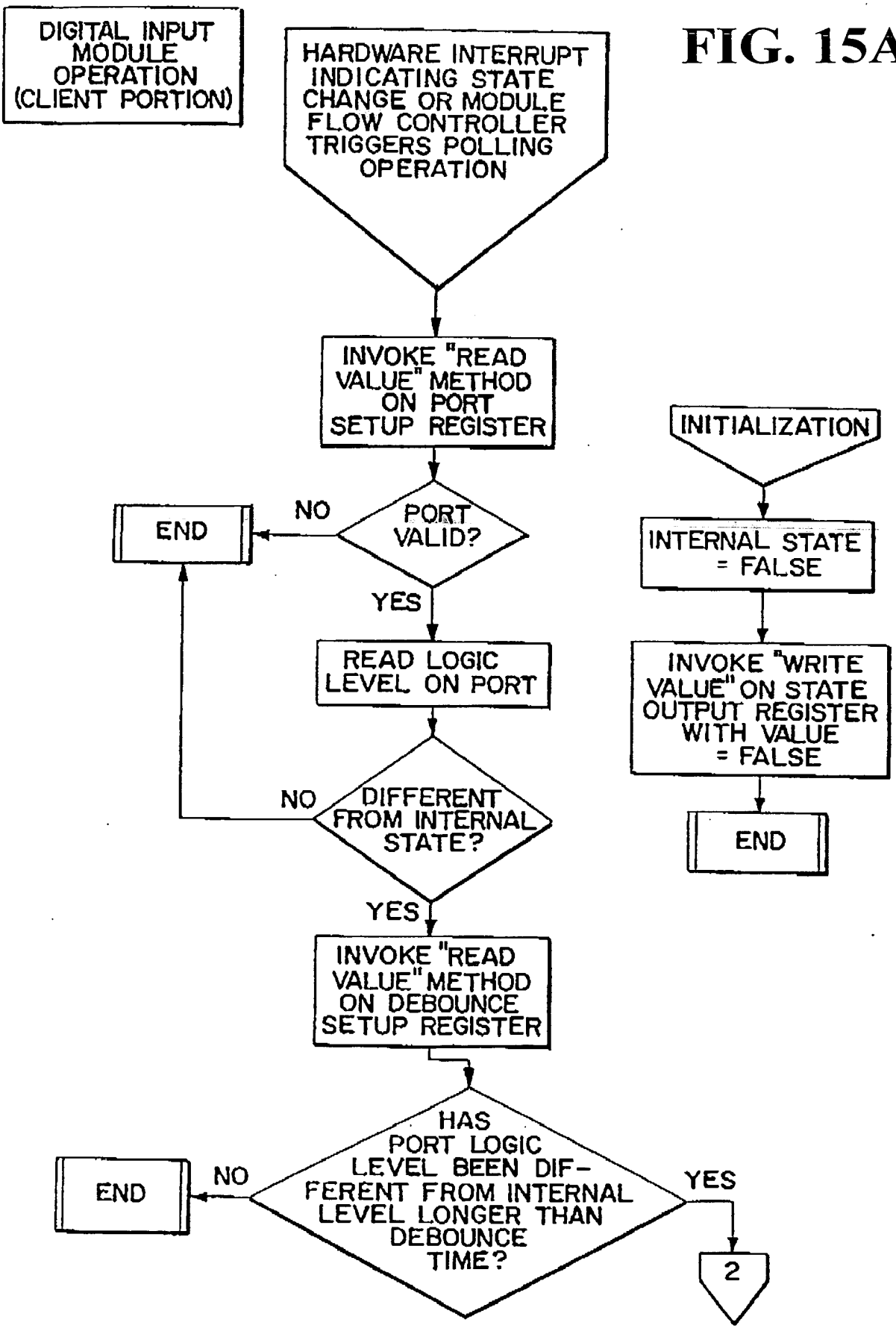

DIGITAL INPUT MODULE OPERATION (CLIENT PORTION)
HARDWARE INTERRUPT INDICATING STATE CHANGE OR MODULE FLOW CONTROLLER TRIGGERS POLLING OPERATION
INVOKE "READ VALUE" METHOD ON PORT SETUP REGISTER
PORT VALID?
NO
END
YES
READ LOGIC LEVEL ON PORT
DIFFERENT FROM INTERNAL STATE?
NO
YES
INVOKE "READ VALUE" METHOD ON DEBOUNCE SETUP REGISTER
HAS PORT LOGIC LEVEL BEEN DIFFERENT FROM INTERNAL LEVEL LONGER THAN DEBOUNCE TIME?
NO
END
YES
2
INITIALIZATION
INTERNAL STATE = FALSE
INVOKE "WRITE VALUE" ON STATE OUTPUT REGISTER WITH VALUE = FALSE
END

FIG. 15B

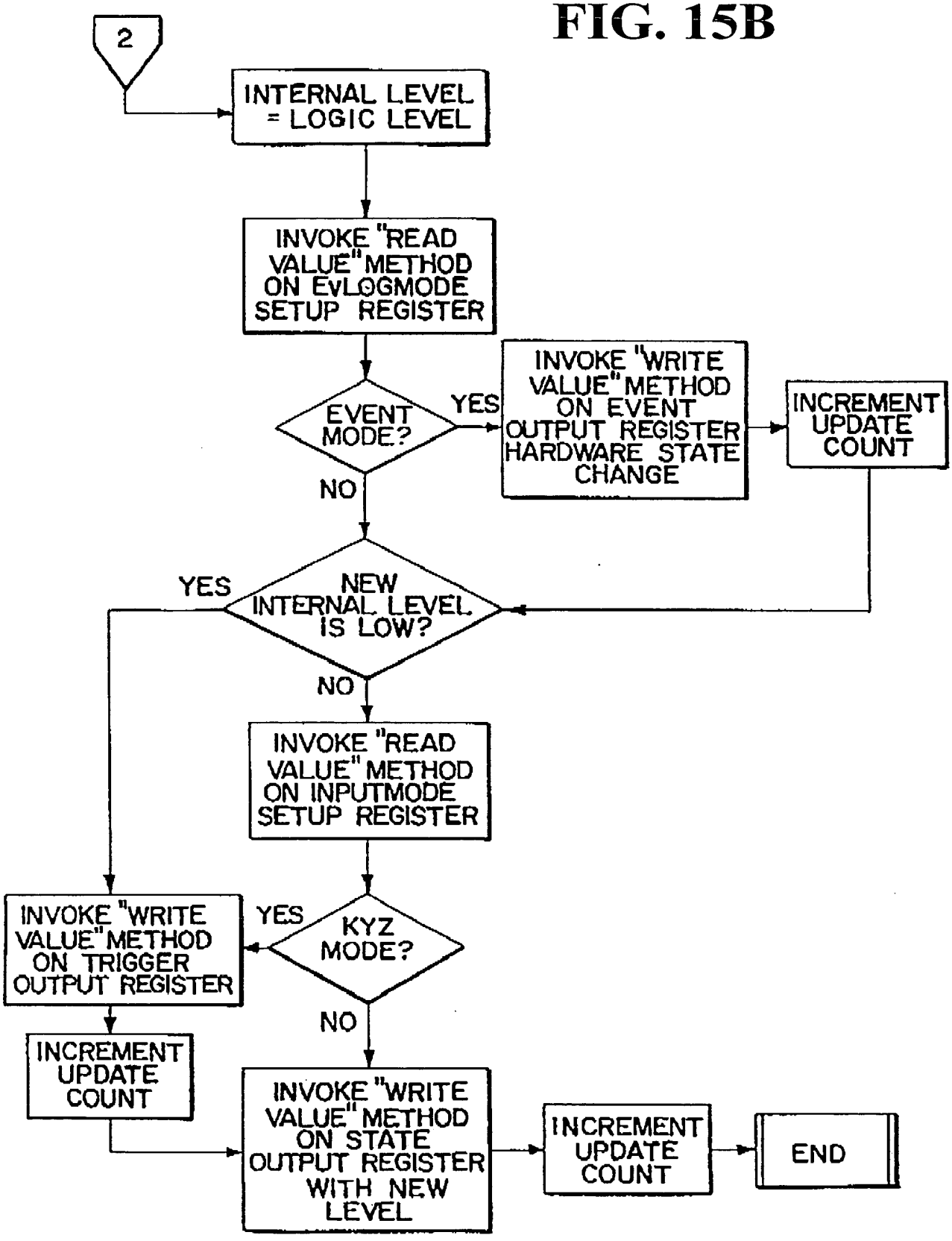

2
INTERNAL LEVEL = LOGIC LEVEL
INVOKE "READ VALUE" METHOD ON EVLOGMODE SETUP REGISTER
EVENT MODE?
YES
INVOKE "WRITE VALUE" METHOD ON EVENT OUTPUT REGISTER HARDWARE STATE CHANGE
INCREMENT UPDATE COUNT
NO
YES
NEW INTERNAL LEVEL IS LOW?
NO
INVOKE "READ VALUE" METHOD ON INPUTMODE SETUP REGISTER
INVOKE "WRITE VALUE" METHOD ON TRIGGER OUTPUT REGISTER
YES
KYZ MODE?
NO
INCREMENT UPDATE COUNT
INVOKE "WRITE VALUE" METHOD ON STATE OUTPUT REGISTER WITH NEW LEVEL
INCREMENT UPDATE COUNT
END

FIG. 16A

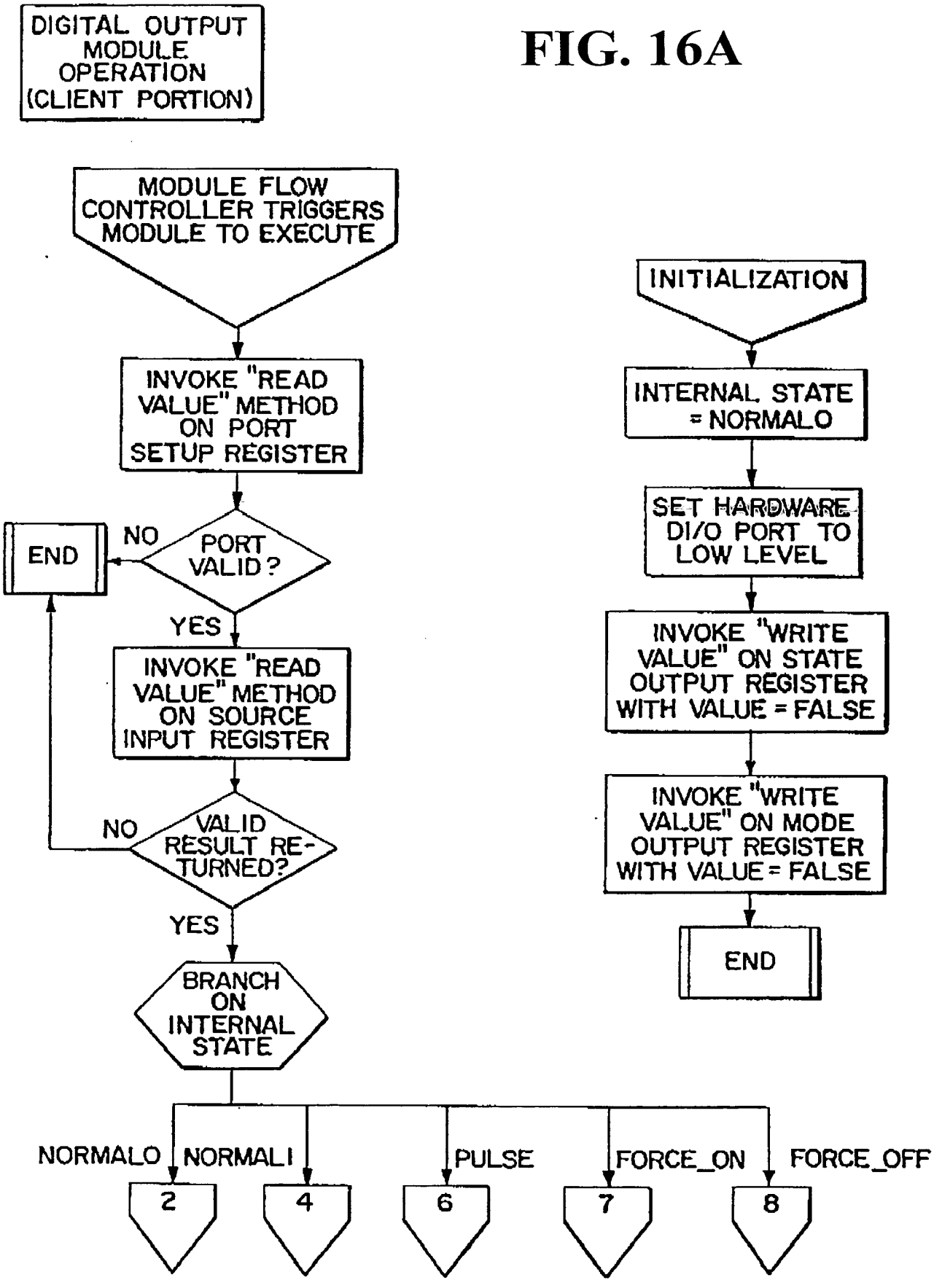

DIGITAL OUTPUT MODULE OPERATION (CLIENT PORTION)
MODULE FLOW CONTROLLER TRIGGERS MODULE TO EXECUTE
INVOKE "READ VALUE" METHOD ON PORT SETUP REGISTER
PORT VALID?
NO
END
YES
INVOKE "READ VALUE" METHOD ON SOURCE INPUT REGISTER
VALID RESULT RE-TURNED?
NO
YES
BRANCH ON INTERNAL STATE
NORMALO
NORMALI
PULSE
FORCE_ON
FORCE_OFF
2
4
6
7
8
INITIALIZATION
INTERNAL STATE = NORMALO
SET HARDWARE DI/O PORT TO LOW LEVEL
INVOKE "WRITE VALUE" ON STATE OUTPUT REGISTER WITH VALUE = FALSE
INVOKE "WRITE VALUE" ON MODE OUTPUT REGISTER WITH VALUE = FALSE
END

FIG. 16B

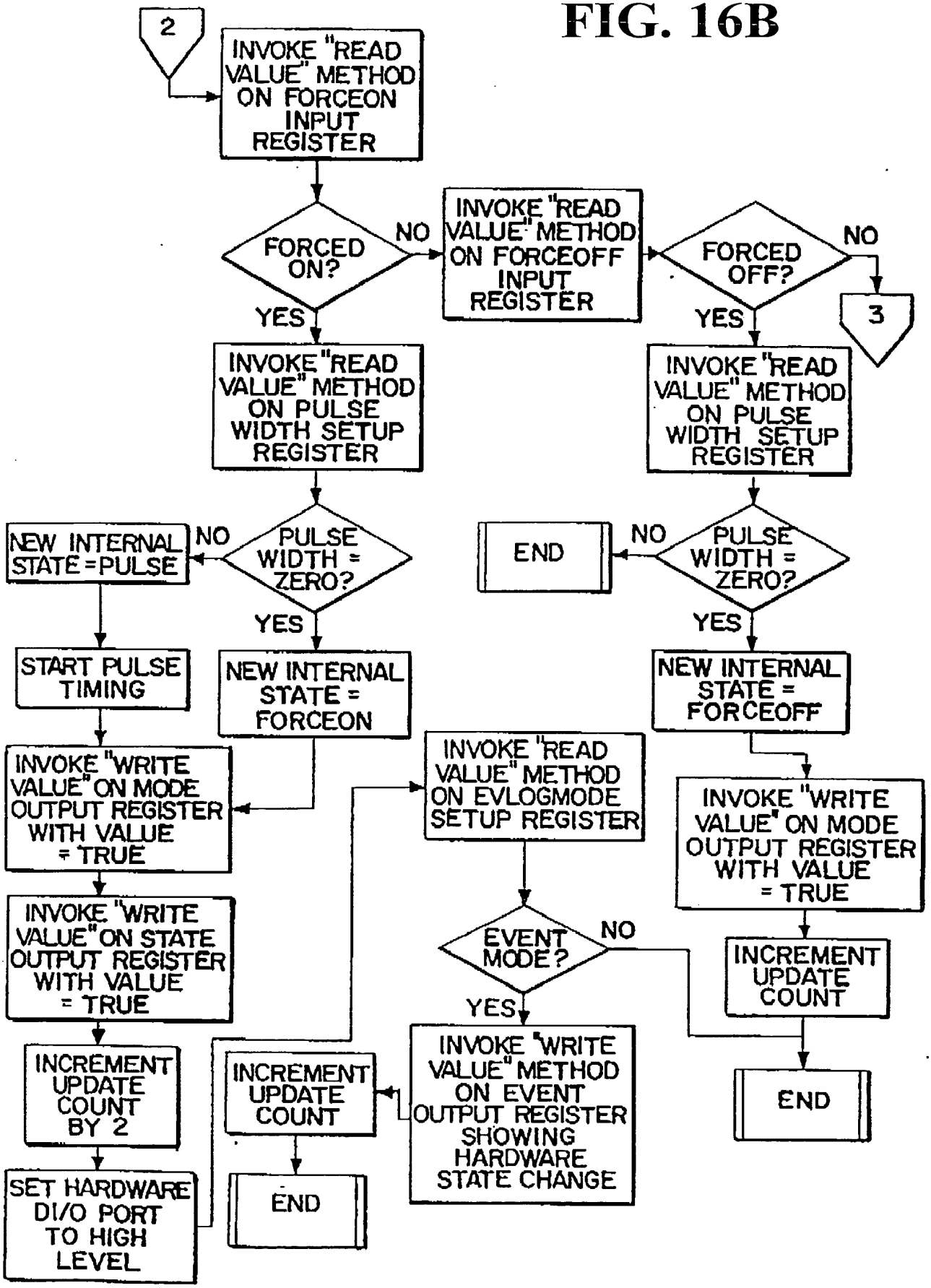

2
INVOKE "READ VALUE" METHOD ON FORCEON INPUT REGISTER
FORCED ON?
NO
YES
INVOKE "READ VALUE" METHOD ON FORCEOFF INPUT REGISTER
FORCED OFF?
NO
YES
3
INVOKE "READ VALUE" METHOD ON PULSE WIDTH SETUP REGISTER
INVOKE "READ VALUE" METHOD ON PULSE WIDTH SETUP REGISTER
NEW INTERNAL STATE = PULSE
NO
PULSE WIDTH = ZERO?
YES
END
NO
PULSE WIDTH = ZERO?
YES
START PULSE TIMING
NEW INTERNAL STATE = FORCEON
NEW INTERNAL STATE = FORCEOFF
INVOKE "WRITE VALUE" ON MODE OUTPUT REGISTER WITH VALUE = TRUE
INVOKE "READ VALUE" METHOD ON EVLOGMODE SETUP REGISTER
INVOKE "WRITE VALUE" ON MODE OUTPUT REGISTER WITH VALUE = TRUE
INVOKE "WRITE VALUE" ON STATE OUTPUT REGISTER WITH VALUE = TRUE
EVENT MODE?
NO
YES
INCREMENT UPDATE COUNT
INCREMENT UPDATE COUNT BY 2
INCREMENT UPDATE COUNT
INVOKE "WRITE VALUE" METHOD ON EVENT OUTPUT REGISTER SHOWING HARDWARE STATE CHANGE
END
SET HARDWARE DI/O PORT TO HIGH LEVEL
END

FIG. 16C

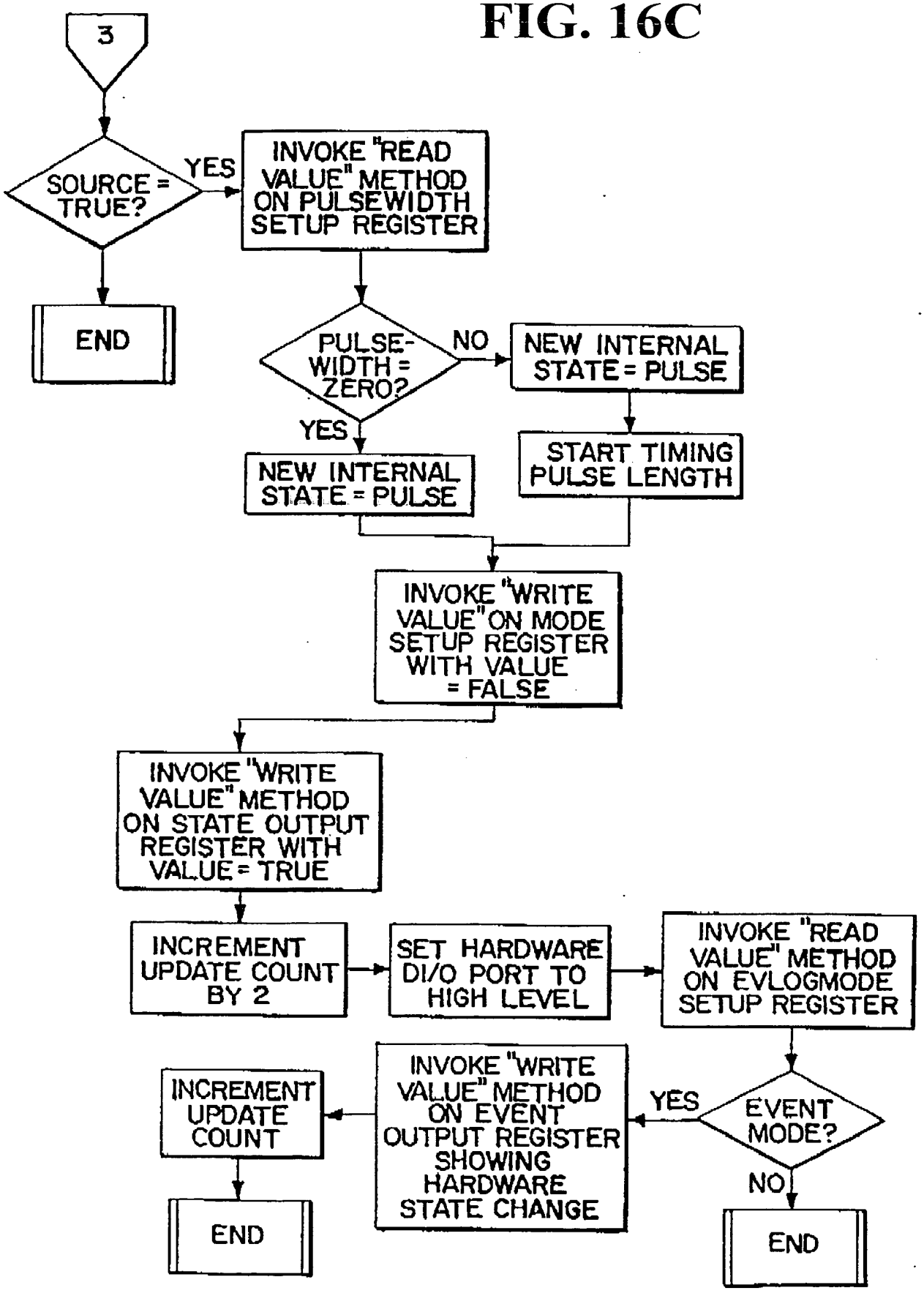

3
SOURCE = TRUE?
YES
INVOKE "READ VALUE" METHOD ON PULSEWIDTH SETUP REGISTER
END
PULSE-WIDTH = ZERO?
NO
NEW INTERNAL STATE = PULSE
YES
START TIMING PULSE LENGTH
NEW INTERNAL STATE = PULSE
INVOKE "WRITE VALUE" ON MODE SETUP REGISTER WITH VALUE = FALSE
INVOKE "WRITE VALUE" METHOD ON STATE OUTPUT REGISTER WITH VALUE = TRUE
INCREMENT UPDATE COUNT BY 2
SET HARDWARE DI/O PORT TO HIGH LEVEL
INVOKE "READ VALUE" METHOD ON EVLOGMODE SETUP REGISTER
EVENT MODE?
YES
NO
INVOKE "WRITE VALUE" METHOD ON EVENT OUTPUT REGISTER SHOWING HARDWARE STATE CHANGE
INCREMENT UPDATE COUNT
END
END

FIG. 16D

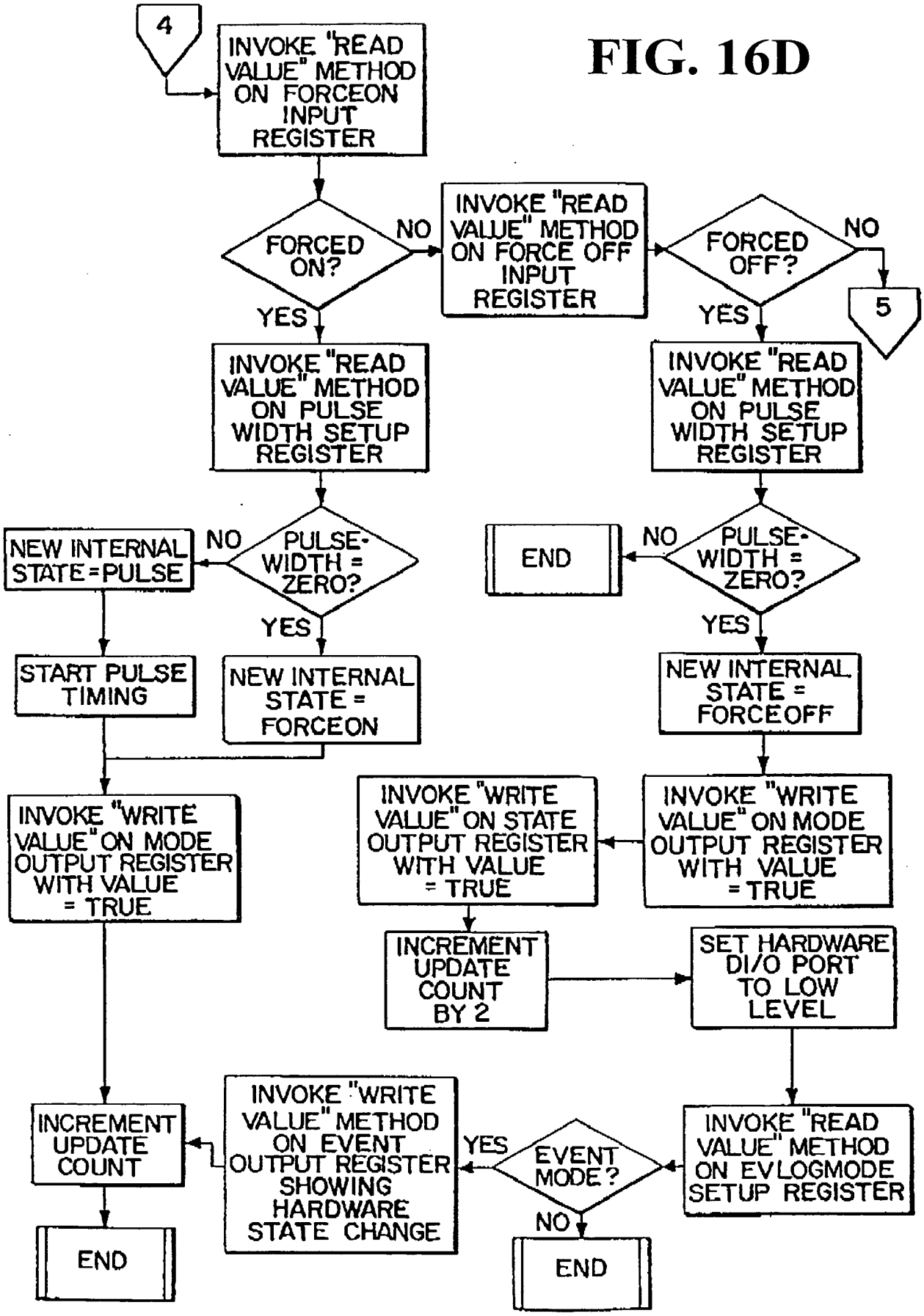

4
INVOKE "READ VALUE" METHOD ON FORCEON INPUT REGISTER
FORCED ON?
NO
YES
INVOKE "READ VALUE" METHOD ON FORCE OFF INPUT REGISTER
FORCED OFF?
NO
YES
5
INVOKE "READ VALUE" METHOD ON PULSE WIDTH SETUP REGISTER
INVOKE "READ VALUE" METHOD ON PULSE WIDTH SETUP REGISTER
NEW INTERNAL STATE=PULSE
NO
PULSE-WIDTH = ZERO?
YES
END
NO
PULSE-WIDTH = ZERO?
YES
START PULSE TIMING
NEW INTERNAL STATE = FORCEON
NEW INTERNAL STATE = FORCEOFF
INVOKE "WRITE VALUE" ON MODE OUTPUT REGISTER WITH VALUE = TRUE
INVOKE "WRITE VALUE" ON STATE OUTPUT REGISTER WITH VALUE = TRUE
INVOKE "WRITE VALUE" ON MODE OUTPUT REGISTER WITH VALUE = TRUE
INCREMENT UPDATE COUNT BY 2
SET HARDWARE DI/O PORT TO LOW LEVEL
INCREMENT UPDATE COUNT
INVOKE "WRITE VALUE" METHOD ON EVENT OUTPUT REGISTER SHOWING HARDWARE STATE CHANGE
YES
EVENT MODE?
NO
INVOKE "READ VALUE" METHOD ON EVLOGMODE SETUP REGISTER
END
END

FIG. 16E

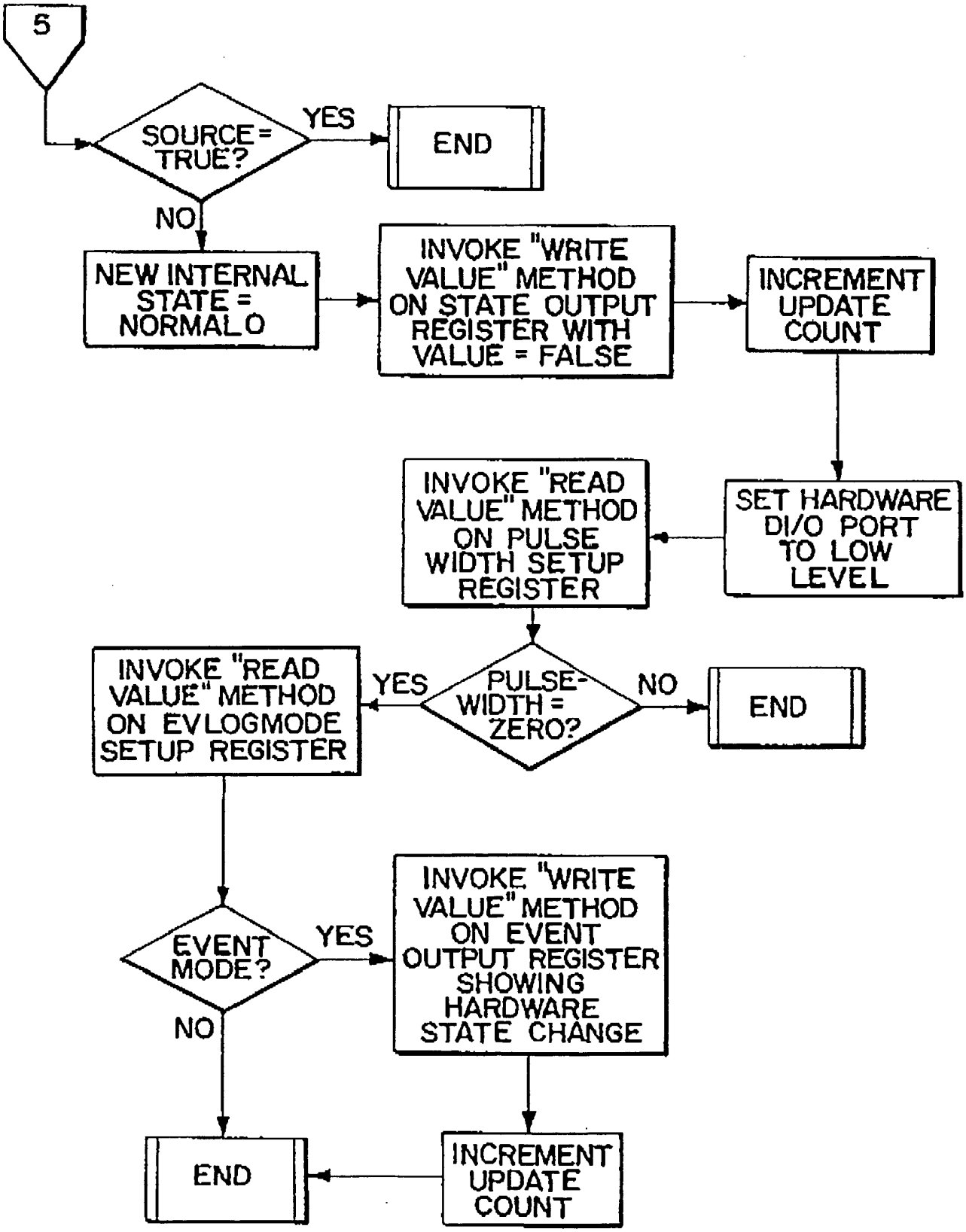

5
SOURCE = TRUE?
YES
END
NO
NEW INTERNAL STATE = NORMAL 0
INVOKE "WRITE VALUE" METHOD ON STATE OUTPUT REGISTER WITH VALUE = FALSE
INCREMENT UPDATE COUNT
SET HARDWARE DI/O PORT TO LOW LEVEL
INVOKE "READ VALUE" METHOD ON PULSE WIDTH SETUP REGISTER
PULSE-WIDTH = ZERO?
YES
NO
END
INVOKE "READ VALUE" METHOD ON EVLOGMODE SETUP REGISTER
EVENT MODE?
YES
NO
INVOKE "WRITE VALUE" METHOD ON EVENT OUTPUT REGISTER SHOWING HARDWARE STATE CHANGE
INCREMENT UPDATE COUNT
END

FIG. 16F

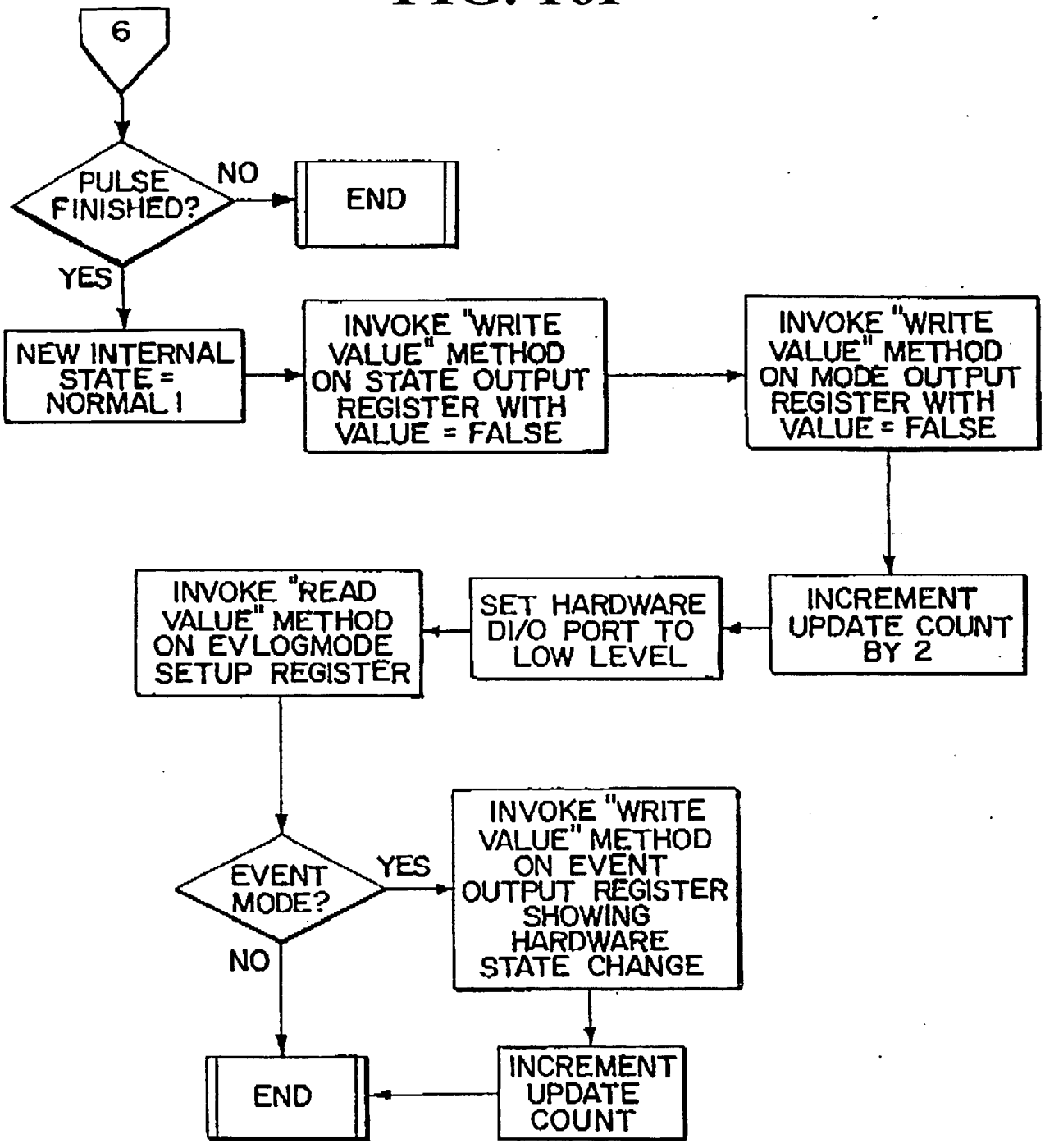

6
PULSE FINISHED?
NO
END
YES
NEW INTERNAL STATE = NORMAL !
INVOKE "WRITE VALUE" METHOD ON STATE OUTPUT REGISTER WITH VALUE = FALSE
INVOKE "WRITE VALUE" METHOD ON MODE OUTPUT REGISTER WITH VALUE = FALSE
INCREMENT UPDATE COUNT BY 2
SET HARDWARE DI/O PORT TO LOW LEVEL
INVOKE "READ VALUE" METHOD ON EVLOGMODE SETUP REGISTER
EVENT MODE?
YES
NO
INVOKE "WRITE VALUE" METHOD ON EVENT OUTPUT REGISTER SHOWING HARDWARE STATE CHANGE
INCREMENT UPDATE COUNT
END

FIG. 16G

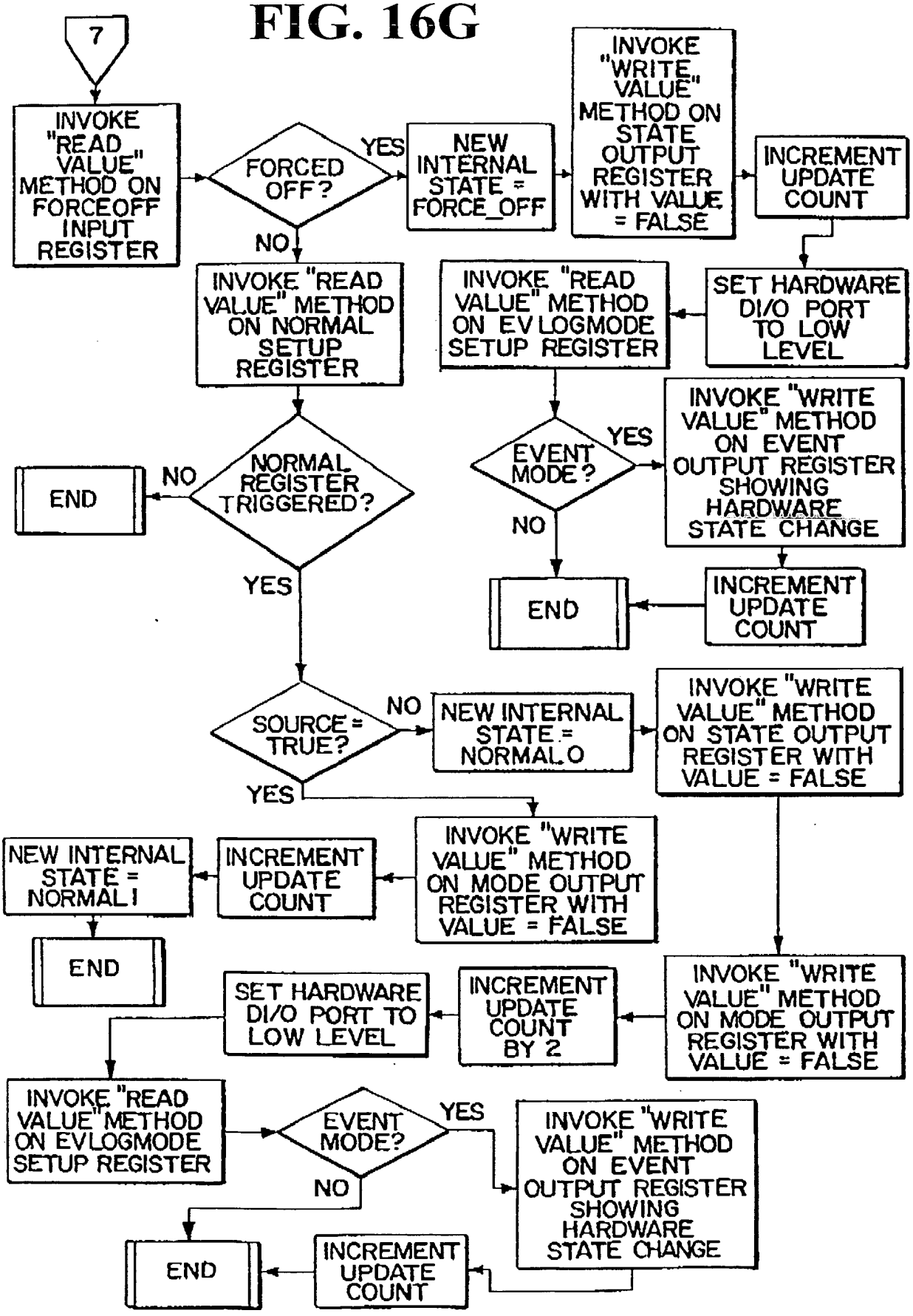

7
INVOKE "READ VALUE" METHOD ON FORCEOFF INPUT REGISTER
FORCED OFF?
YES
NO
NEW INTERNAL STATE = FORCE_OFF
INVOKE "WRITE VALUE" METHOD ON STATE OUTPUT REGISTER WITH VALUE = FALSE
INCREMENT UPDATE COUNT
SET HARDWARE DI/O PORT TO LOW LEVEL
INVOKE "READ VALUE" METHOD ON EVLOGMODE SETUP REGISTER
INVOKE "READ VALUE" METHOD ON NORMAL SETUP REGISTER
EVENT MODE?
YES
NO
INVOKE "WRITE VALUE" METHOD ON EVENT OUTPUT REGISTER SHOWING HARDWARE STATE CHANGE
INCREMENT UPDATE COUNT
END
NORMAL REGISTER TRIGGERED?
NO
YES
END
SOURCE = TRUE?
NO
YES
NEW INTERNAL STATE = NORMAL.O
INVOKE "WRITE VALUE" METHOD ON STATE OUTPUT REGISTER WITH VALUE = FALSE
INVOKE "WRITE VALUE" METHOD ON MODE OUTPUT REGISTER WITH VALUE = FALSE
INCREMENT UPDATE COUNT
NEW INTERNAL STATE = NORMAL.I
END
INVOKE "WRITE VALUE" METHOD ON MODE OUTPUT REGISTER WITH VALUE = FALSE
INCREMENT UPDATE COUNT BY 2
SET HARDWARE DI/O PORT TO LOW LEVEL
INVOKE "READ VALUE" METHOD ON EVLOGMODE SETUP REGISTER
EVENT MODE?
YES
NO
INVOKE "WRITE VALUE" METHOD ON EVENT OUTPUT REGISTER SHOWING HARDWARE STATE CHANGE
INCREMENT UPDATE COUNT
END

FIG. 16H

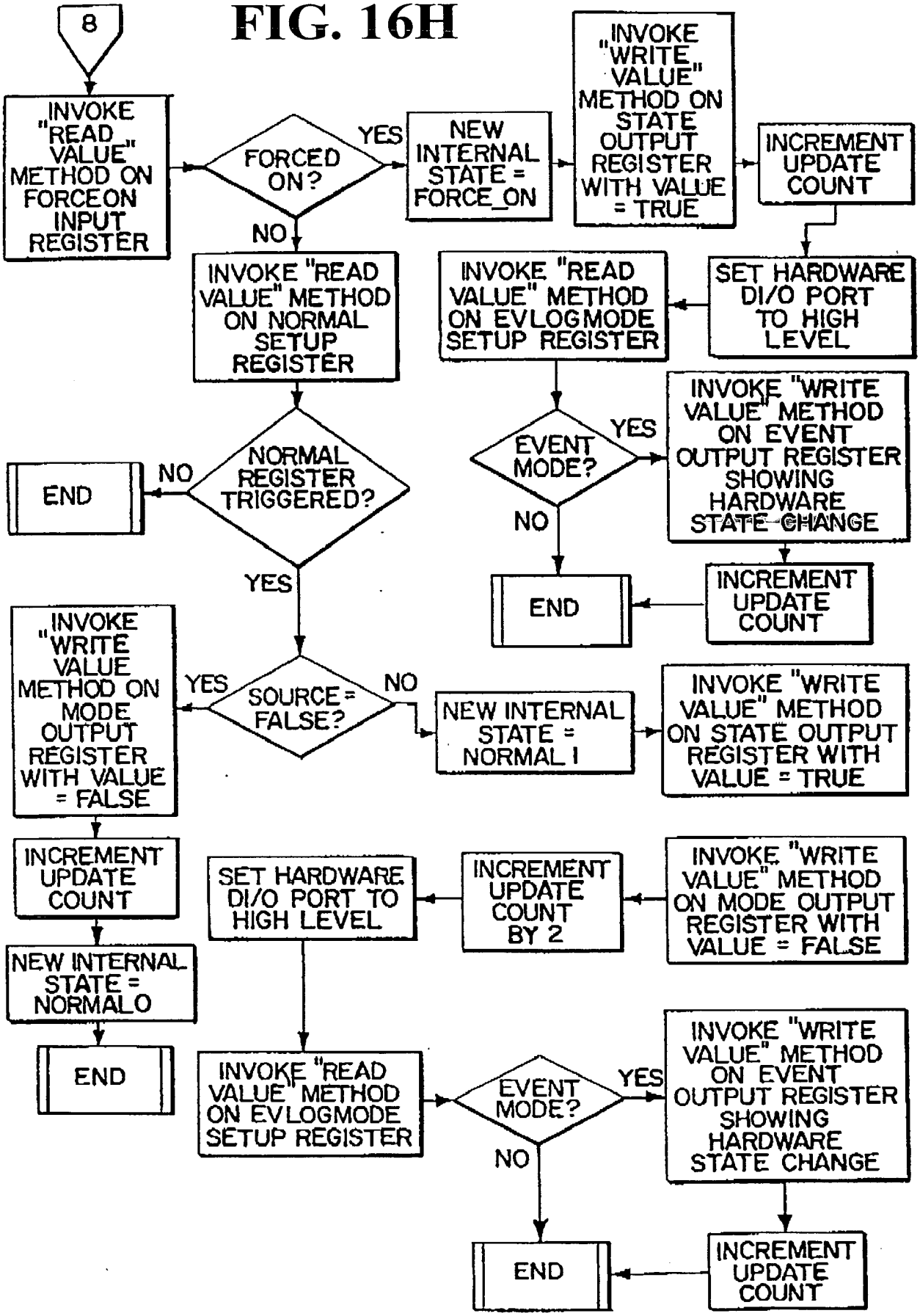

8
INVOKE "READ VALUE" METHOD ON FORCEON INPUT REGISTER
FORCED ON?
YES
NEW INTERNAL STATE = FORCE_ON
INVOKE "WRITE VALUE" METHOD ON STATE OUTPUT REGISTER WITH VALUE = TRUE
INCREMENT UPDATE COUNT
NO
INVOKE "READ VALUE" METHOD ON NORMAL SETUP REGISTER
INVOKE "READ VALUE" METHOD ON EVLOGMODE SETUP REGISTER
SET HARDWARE DI/O PORT TO HIGH LEVEL
NORMAL REGISTER TRIGGERED?
NO
END
EVENT MODE?
YES
INVOKE "WRITE VALUE" METHOD ON EVENT OUTPUT REGISTER SHOWING HARDWARE STATE CHANGE
NO
END
INCREMENT UPDATE COUNT
YES
INVOKE "WRITE VALUE METHOD ON MODE OUTPUT REGISTER WITH VALUE = FALSE
SOURCE = FALSE?
YES
NO
NEW INTERNAL STATE = NORMAL 1
INVOKE "WRITE VALUE" METHOD ON STATE OUTPUT REGISTER WITH VALUE = TRUE
INCREMENT UPDATE COUNT
SET HARDWARE DI/O PORT TO HIGH LEVEL
INCREMENT UPDATE COUNT BY 2
INVOKE "WRITE VALUE" METHOD ON MODE OUTPUT REGISTER WITH VALUE = FALSE
NEW INTERNAL STATE = NORMAL 0
END
INVOKE "READ VALUE" METHOD ON EVLOGMODE SETUP REGISTER
EVENT MODE?
YES
INVOKE "WRITE VALUE" METHOD ON EVENT OUTPUT REGISTER SHOWING HARDWARE STATE CHANGE
NO
END
INCREMENT UPDATE COUNT

FIG. 17

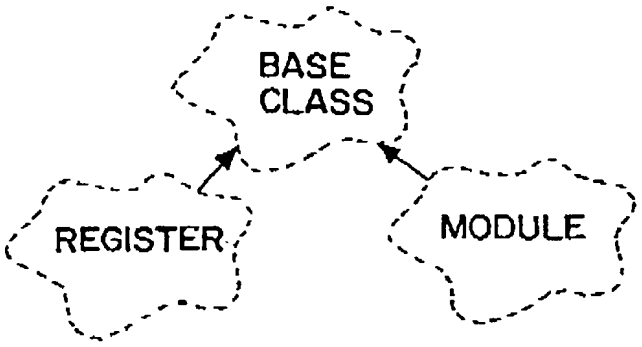


FIG. 17A

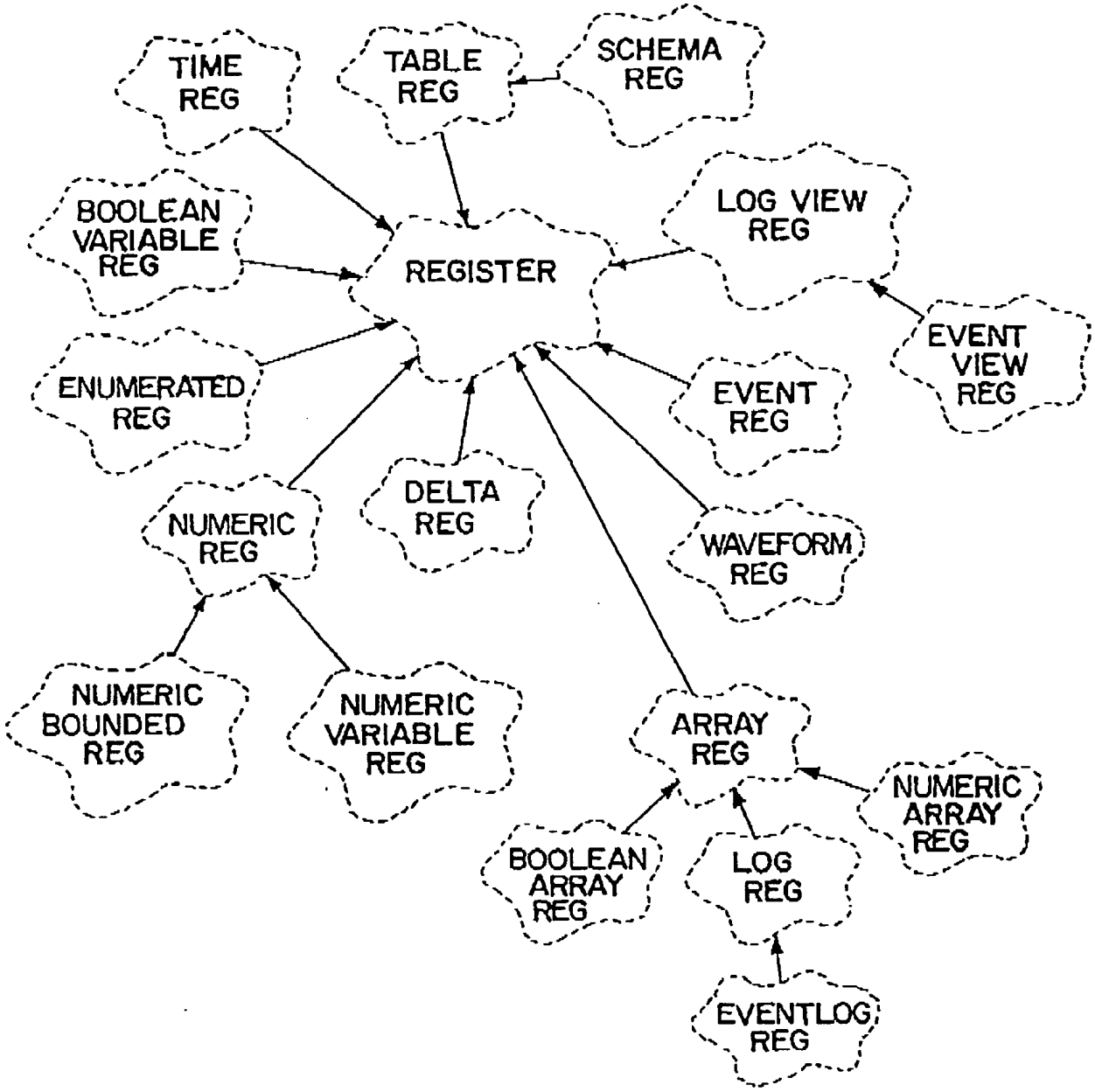


FIG. 17B

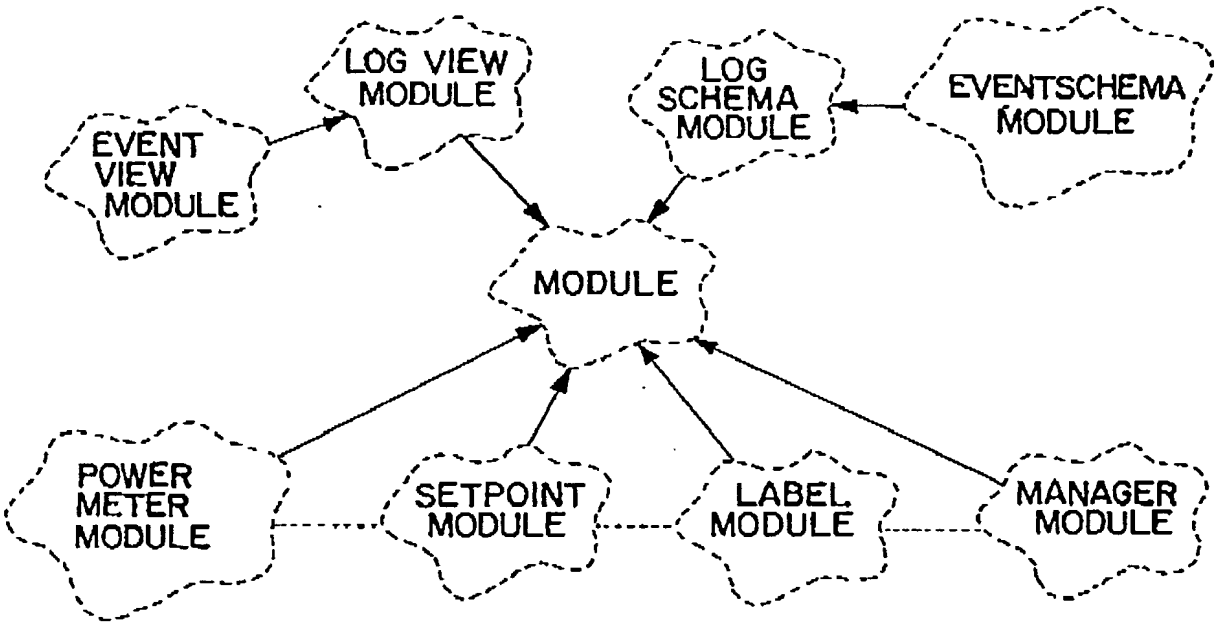


FIG. 17C

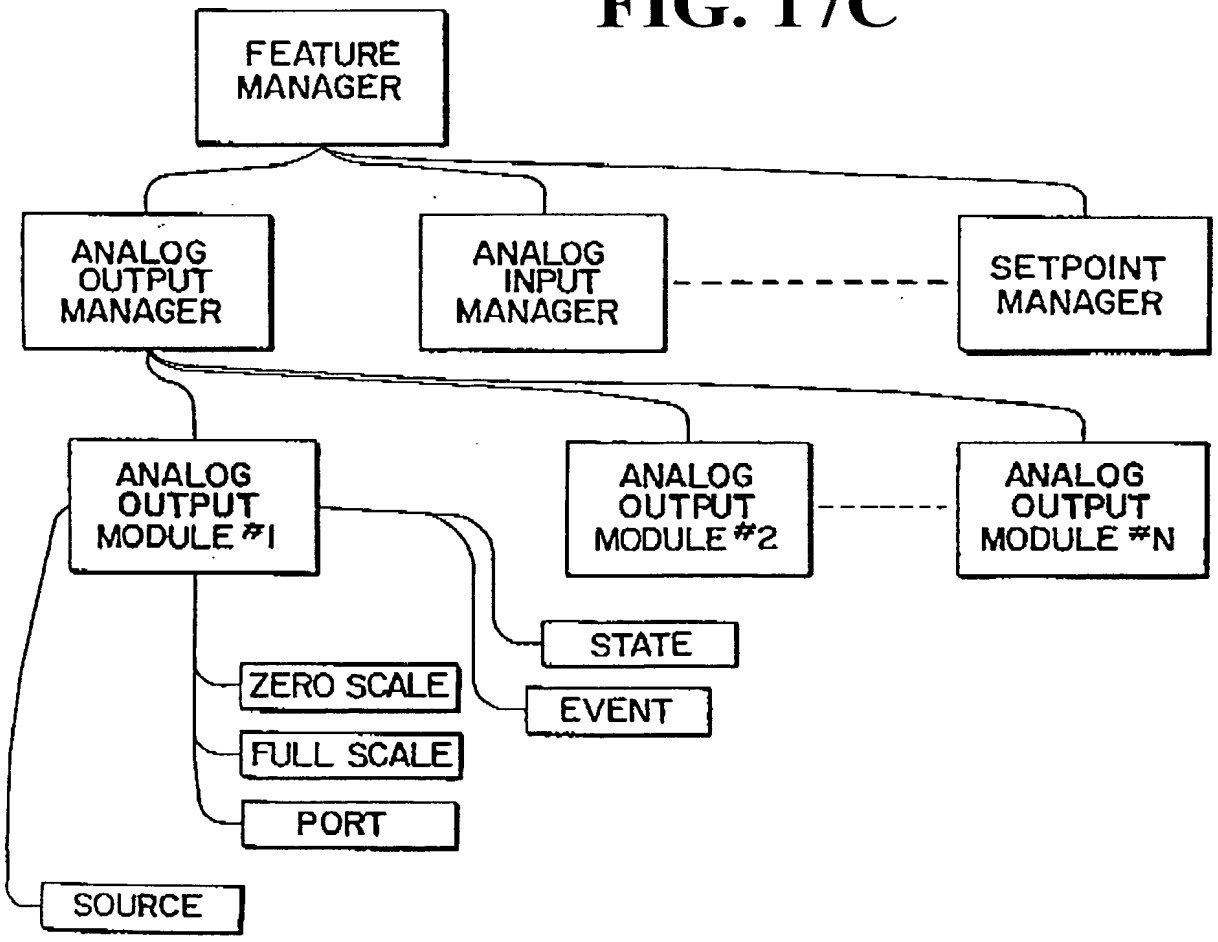


FIG. 18

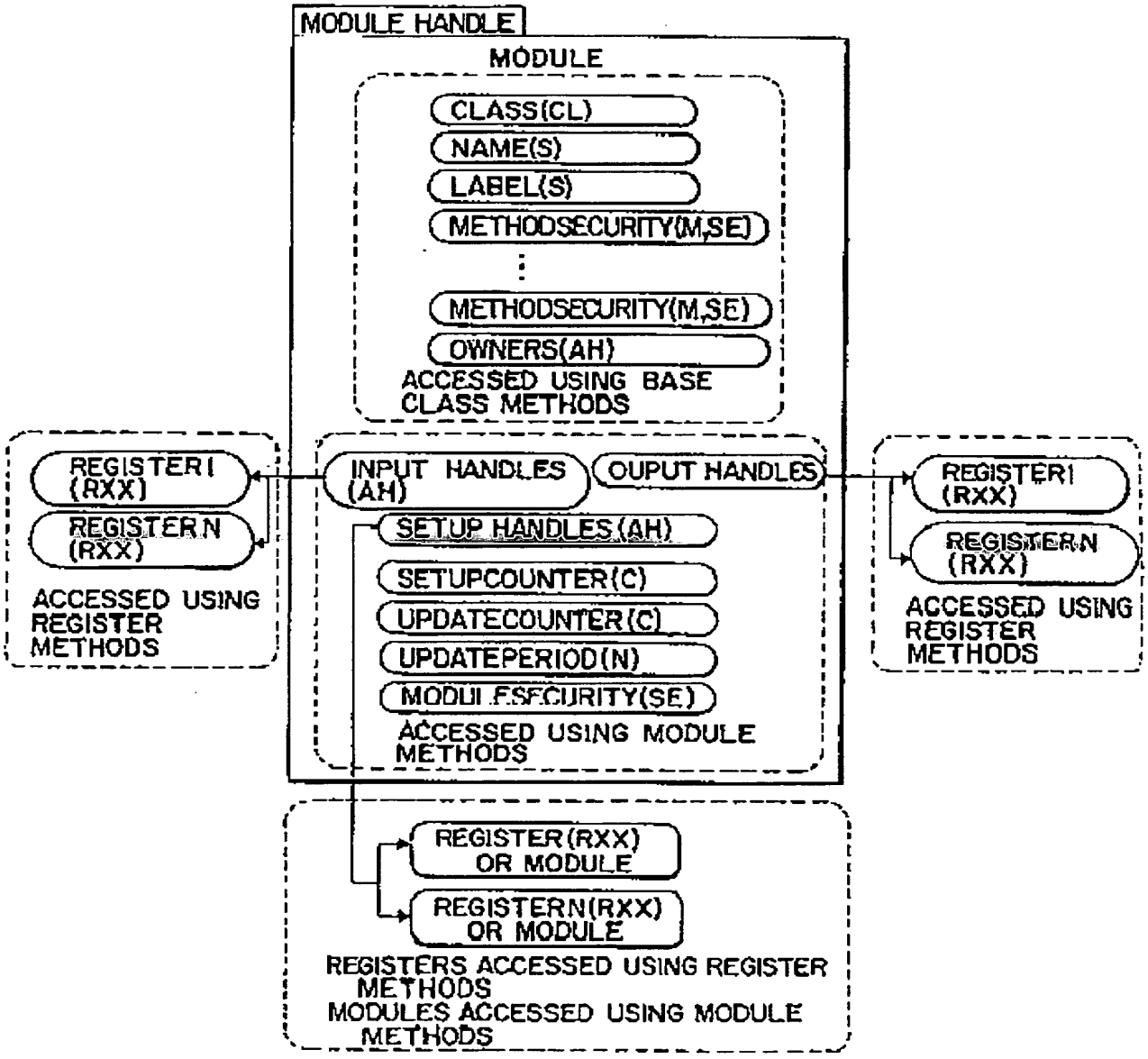

MODULE HANDLE
MODULE
CLASS(CL)
NAME(S)
LABEL(S)
METHODSECURITY(M,SE)
METHODSECURITY(M,SE)
OWNERS(AH)
ACCESSED USING BASE CLASS METHODS
REGISTER1 (RXX)
REGISTERN (RXX)
ACCESSED USING REGISTER METHODS
INPUT HANDLES (AH)
OUPUT HANDLES
SETUP HANDLES(AH)
SETUPCOUNTER(C)
UPDATECOUNTER(C)
UPDATEPERIOD(N)
MODULESECURITY(SE)
ACCESSED USING MODULE METHODS
REGISTER1 (RXX)
REGISTERN (RXX)
ACCESSED USING REGISTER METHODS
REGISTER(RXX) OR MODULE
REGISTERN(RXX) OR MODULE
REGISTERS ACCESSED USING REGISTER METHODS
MODULES ACCESSED USING MODULE METHODS

FIG. 19

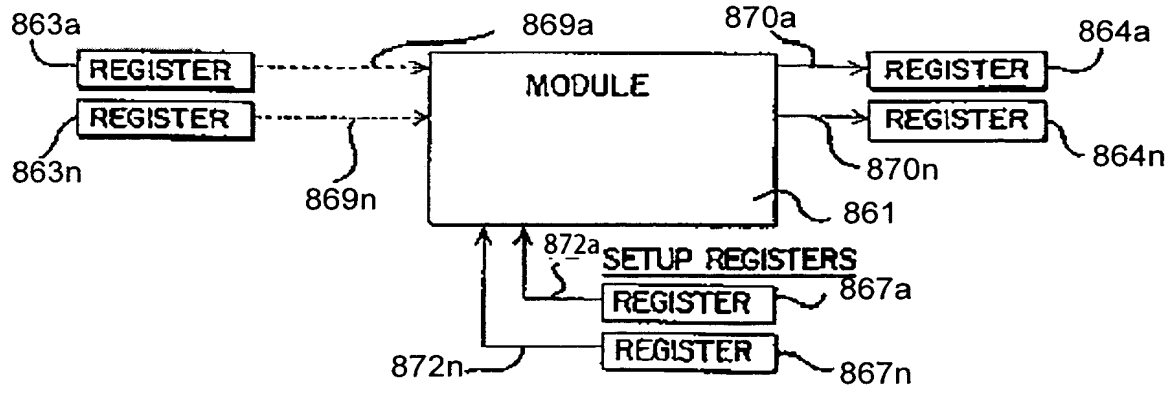

863a
REGISTER
REGISTER
863n
869a
869n
MODULE
870a
REGISTER
REGISTER
870n
864a
864n
861
872a
SETUP REGISTERS
REGISTER
867a
REGISTER
872n
867n

FIG. 19A

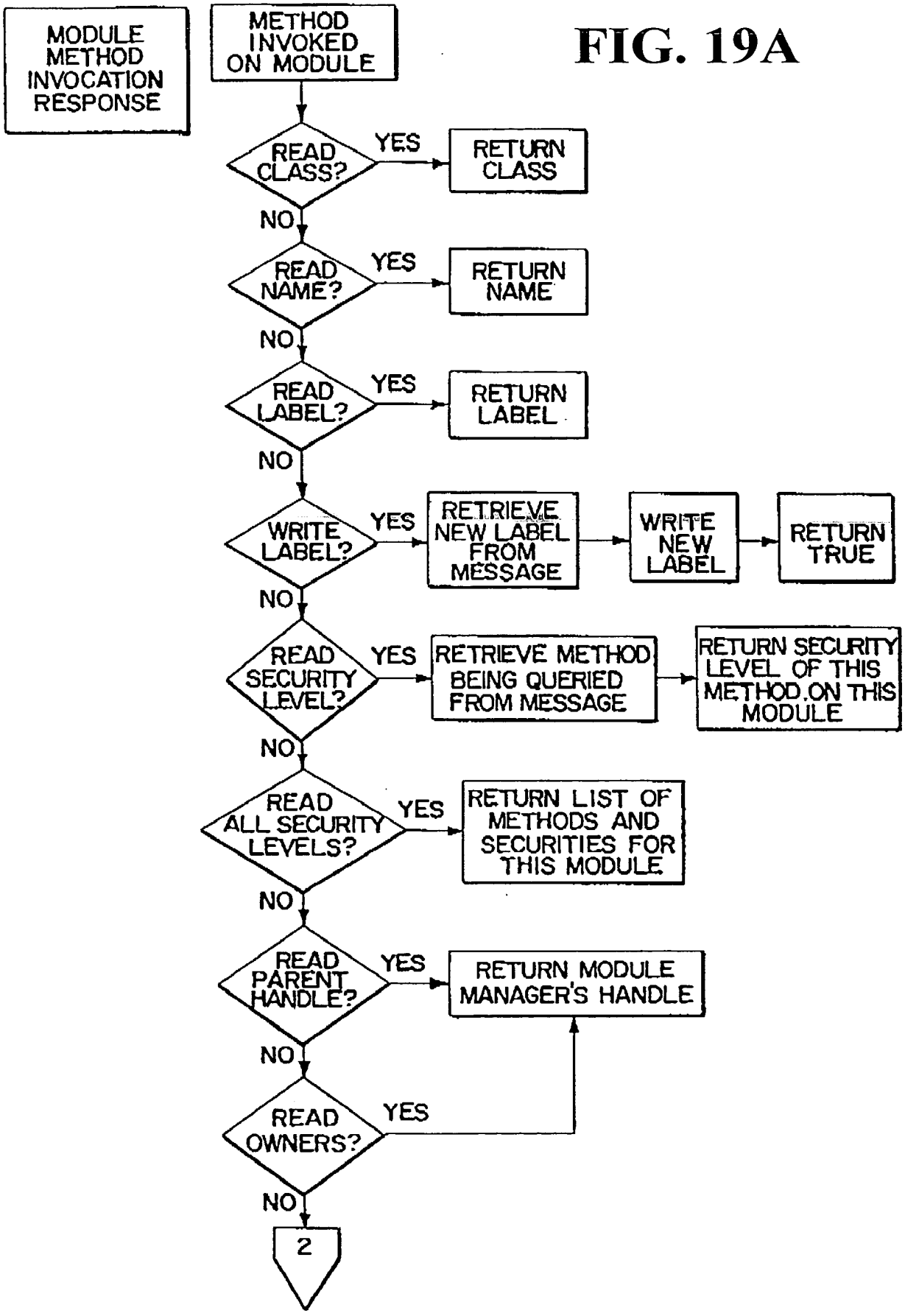

MODULE METHOD INVOCATION RESPONSE
METHOD INVOKED ON MODULE
READ CLASS?
YES
RETURN CLASS
NO
READ NAME?
YES
RETURN NAME
NO
READ LABEL?
YES
RETURN LABEL
NO
WRITE LABEL?
YES
RETRIEVE NEW LABEL FROM MESSAGE
WRITE NEW LABEL
RETURN TRUE
NO
READ SECURITY LEVEL?
YES
RETRIEVE METHOD BEING QUERIED FROM MESSAGE
RETURN SECURITY LEVEL OF THIS METHOD ON THIS MODULE
NO
READ ALL SECURITY LEVELS?
YES
RETURN LIST OF METHODS AND SECURITIES FOR THIS MODULE
NO
READ PARENT HANDLE?
YES
RETURN MODULE MANAGER'S HANDLE
NO
READ OWNERS?
YES
NO
2

FIG. 19B

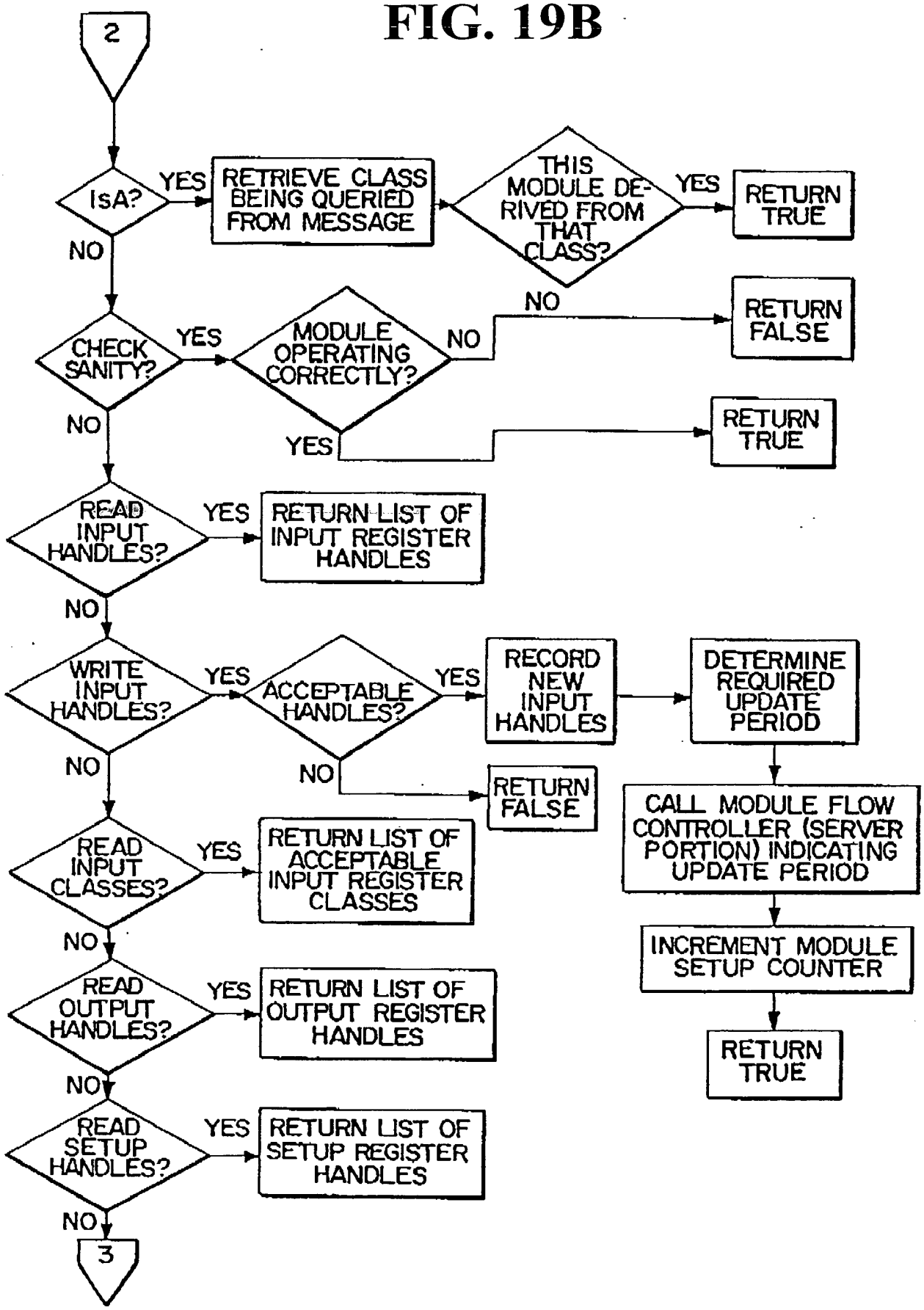

2
IsA?
YES
RETRIEVE CLASS BEING QUERIED FROM MESSAGE
THIS MODULE DERIVED FROM THAT CLASS?
YES
RETURN TRUE
NO
CHECK SANITY?
YES
MODULE OPERATING CORRECTLY?
NO
NO
RETURN FALSE
YES
RETURN TRUE
NO
READ INPUT HANDLES?
YES
RETURN LIST OF INPUT REGISTER HANDLES
NO
WRITE INPUT HANDLES?
YES
ACCEPTABLE HANDLES?
YES
RECORD NEW INPUT HANDLES
DETERMINE REQUIRED UPDATE PERIOD
NO
NO
RETURN FALSE
CALL MODULE FLOW CONTROLLER (SERVER PORTION) INDICATING UPDATE PERIOD
READ INPUT CLASSES?
YES
RETURN LIST OF ACCEPTABLE INPUT REGISTER CLASSES
NO
INCREMENT MODULE SETUP COUNTER
READ OUTPUT HANDLES?
YES
RETURN LIST OF OUTPUT REGISTER HANDLES
NO
RETURN TRUE
READ SETUP HANDLES?
YES
RETURN LIST OF SETUP REGISTER HANDLES
NO
3

FIG. 19C

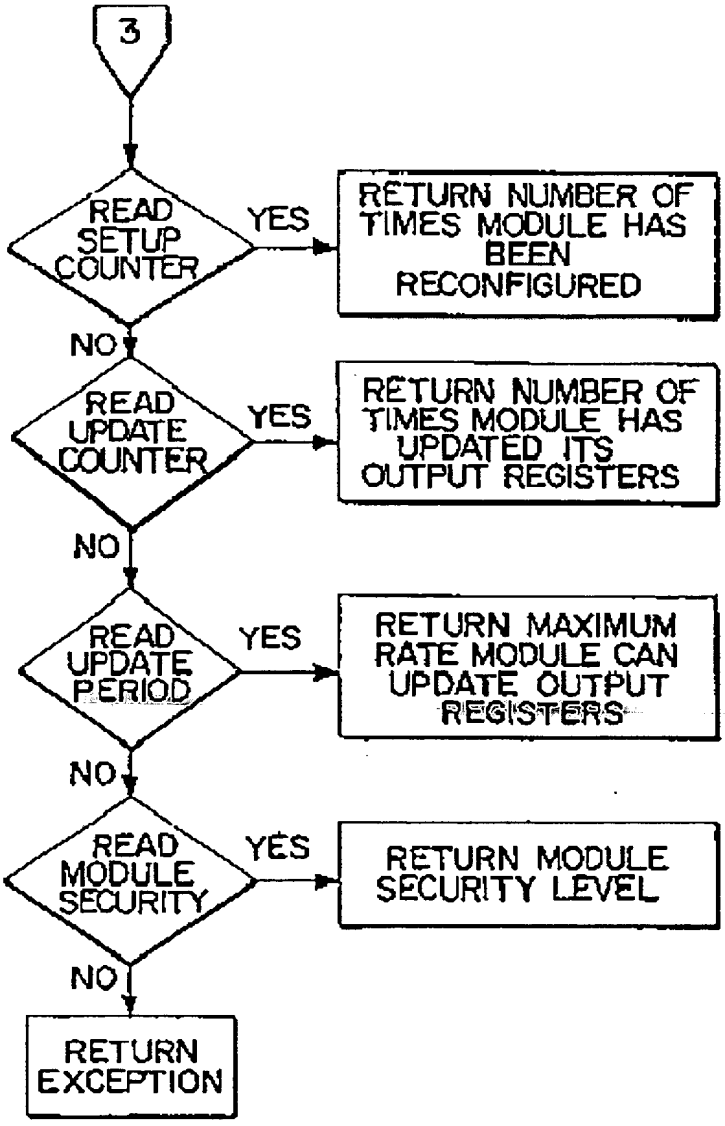


FIG. 20

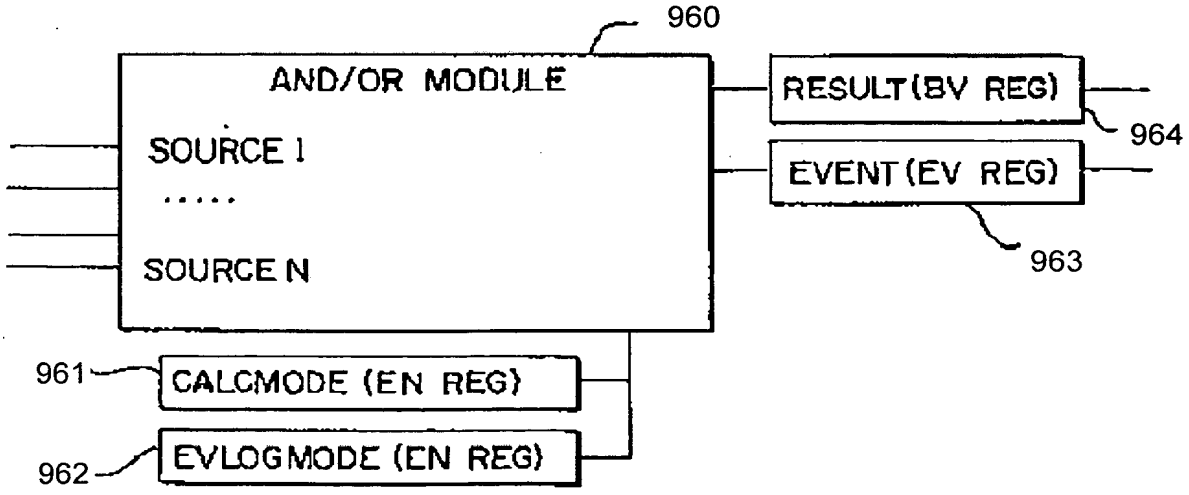


FIG. 20A

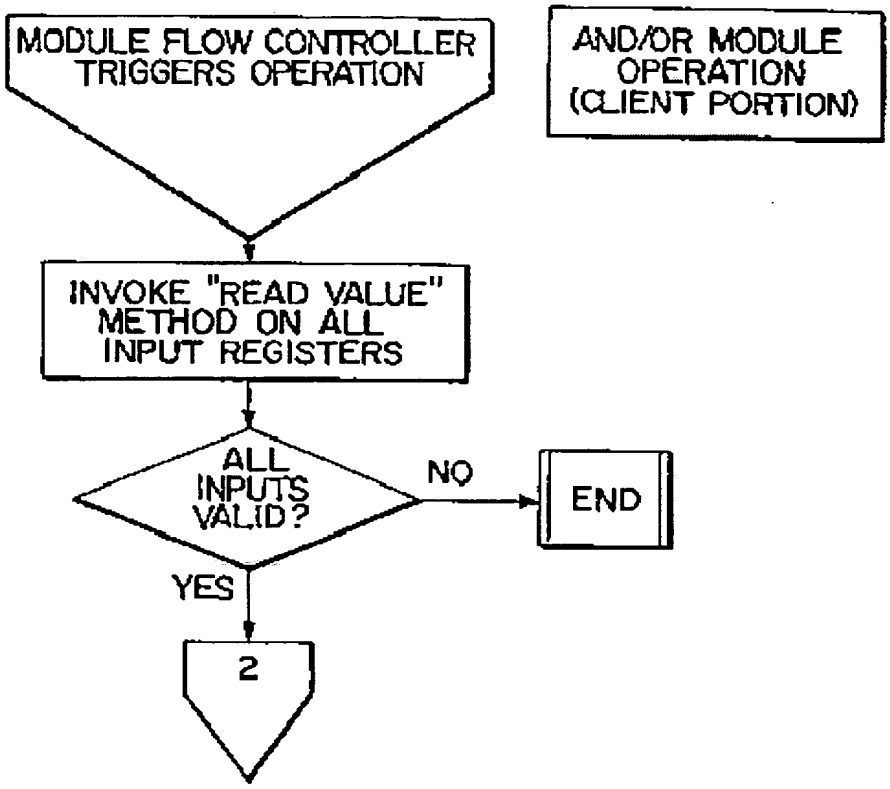


FIG. 21

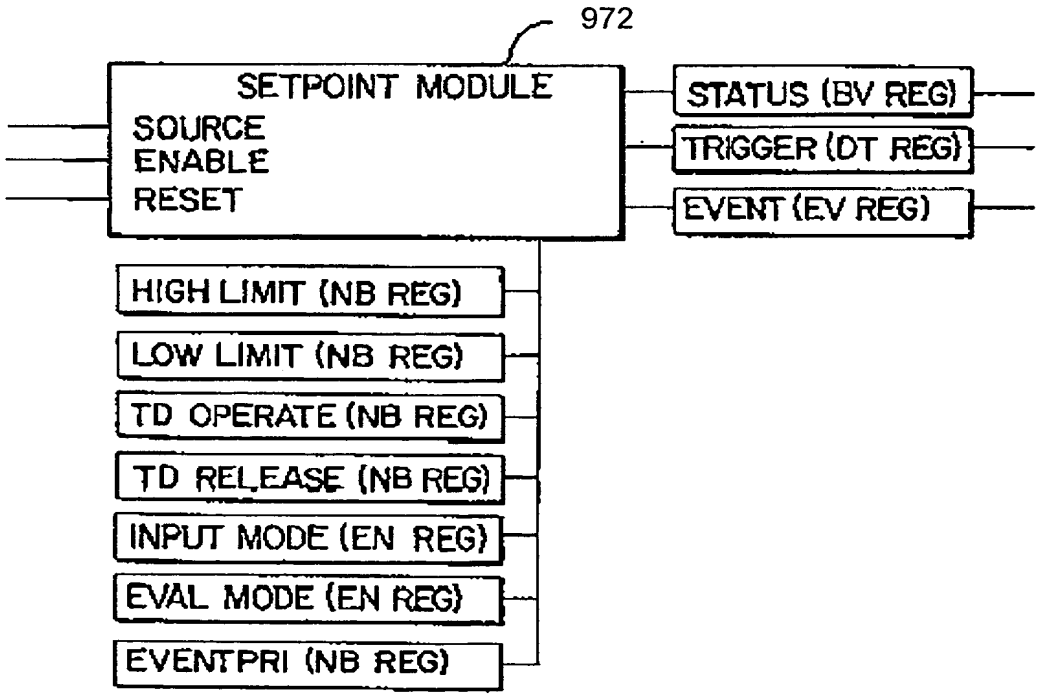


FIG. 20B

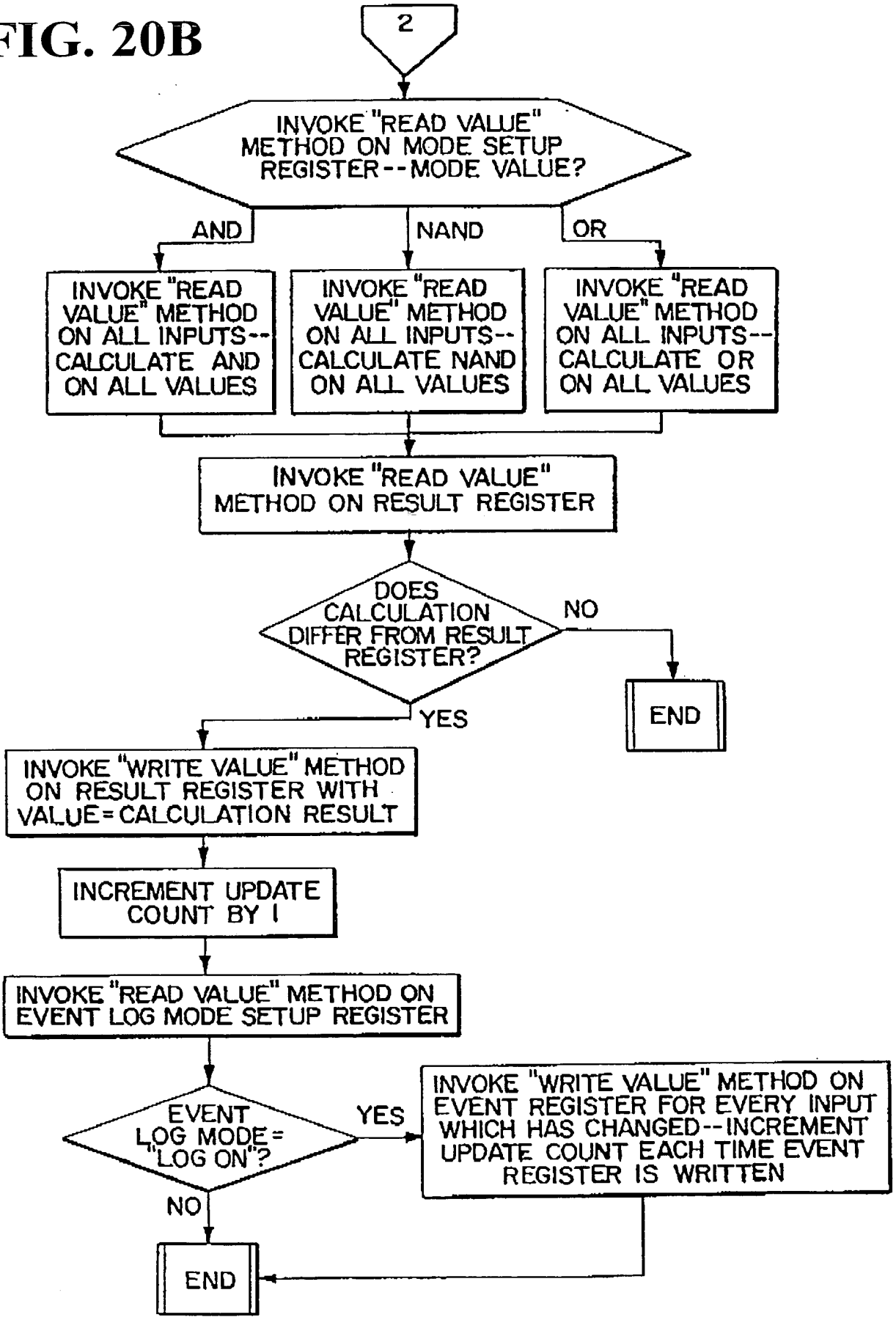

2
INVOKE "READ VALUE" METHOD ON MODE SETUP REGISTER--MODE VALUE?
AND
NAND
OR
INVOKE "READ VALUE" METHOD ON ALL INPUTS--CALCULATE AND ON ALL VALUES
INVOKE "READ VALUE" METHOD ON ALL INPUTS--CALCULATE NAND ON ALL VALUES
INVOKE "READ VALUE" METHOD ON ALL INPUTS--CALCULATE OR ON ALL VALUES
INVOKE "READ VALUE" METHOD ON RESULT REGISTER
DOES CALCULATION DIFFER FROM RESULT REGISTER?
NO
YES
END
INVOKE "WRITE VALUE" METHOD ON RESULT REGISTER WITH VALUE=CALCULATION RESULT
INCREMENT UPDATE COUNT BY 1
INVOKE "READ VALUE" METHOD ON EVENT LOG MODE SETUP REGISTER
EVENT LOG MODE="LOG ON"?
YES
INVOKE "WRITE VALUE" METHOD ON EVENT REGISTER FOR EVERY INPUT WHICH HAS CHANGED--INCREMENT UPDATE COUNT EACH TIME EVENT REGISTER IS WRITTEN
NO
END

FIG. 21A

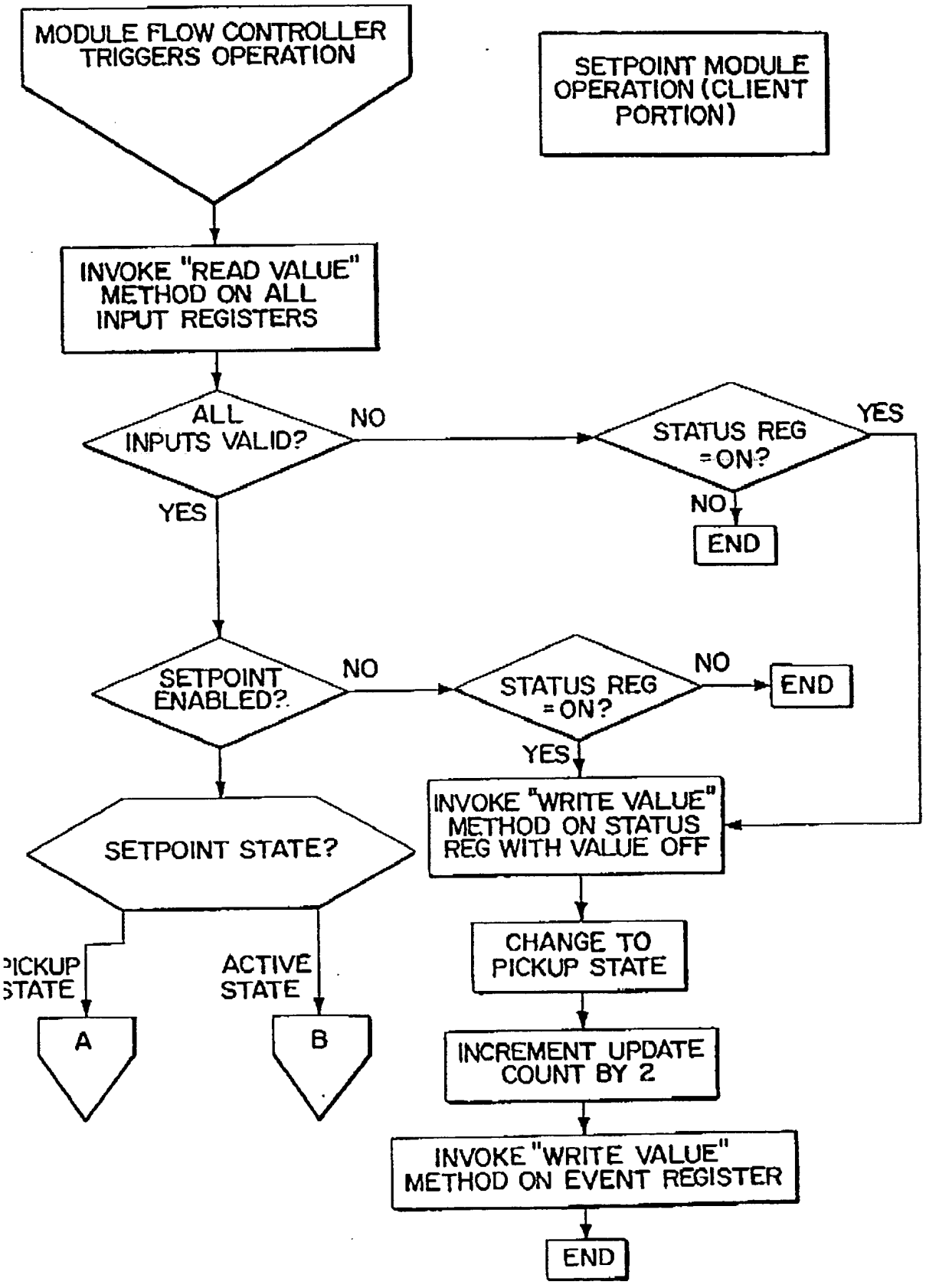

MODULE FLOW CONTROLLER TRIGGERS OPERATION
SETPOINT MODULE OPERATION (CLIENT PORTION)
INVOKE "READ VALUE" METHOD ON ALL INPUT REGISTERS
ALL INPUTS VALID?
NO
YES
STATUS REG =ON?
YES
NO
END
SETPOINT ENABLED?
NO
STATUS REG =ON?
NO
END
YES
SETPOINT STATE?
INVOKE "WRITE VALUE" METHOD ON STATUS REG WITH VALUE OFF
PICKUP STATE
ACTIVE STATE
CHANGE TO PICKUP STATE
A
B
INCREMENT UPDATE COUNT BY 2
INVOKE "WRITE VALUE" METHOD ON EVENT REGISTER
END

FIG. 21B

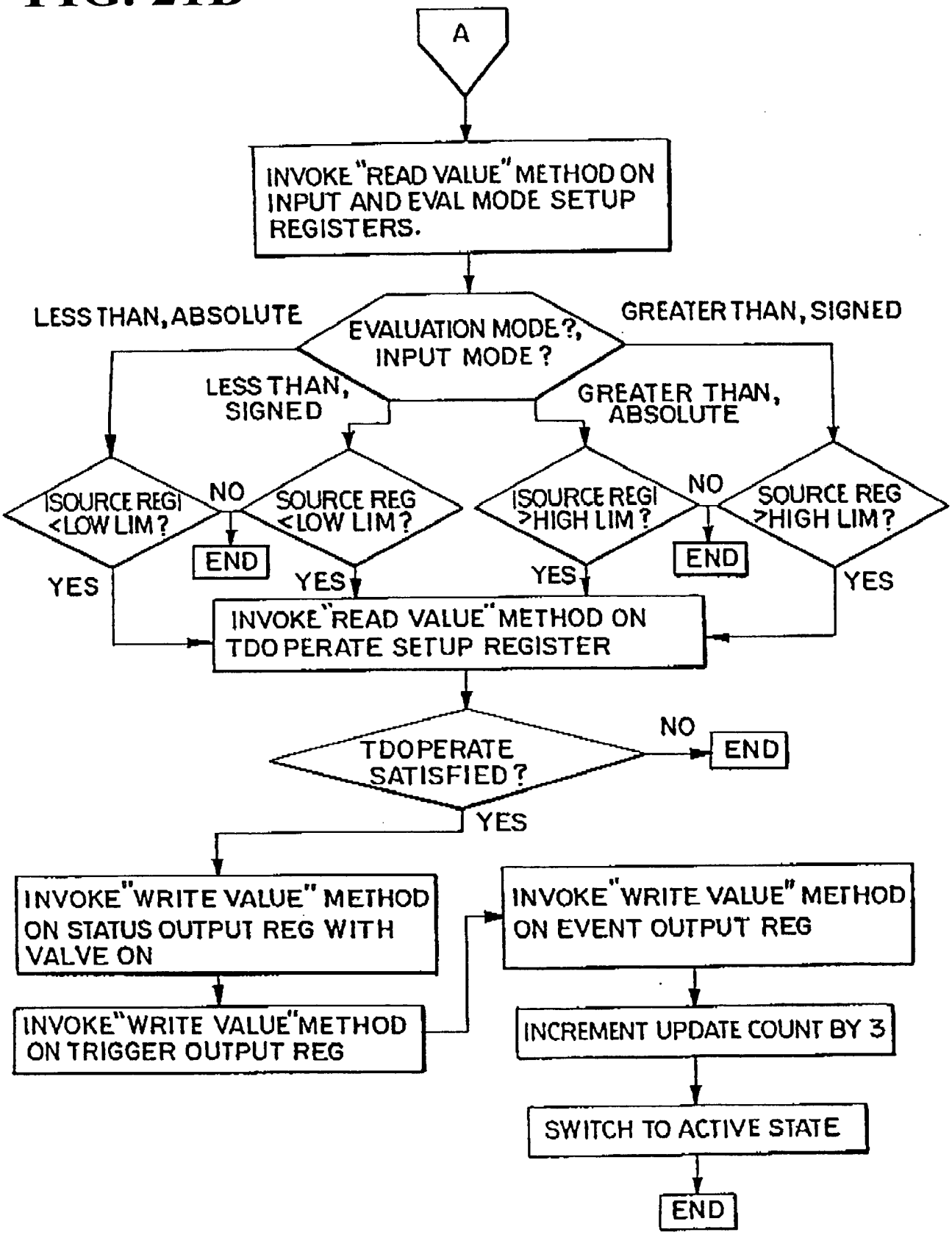
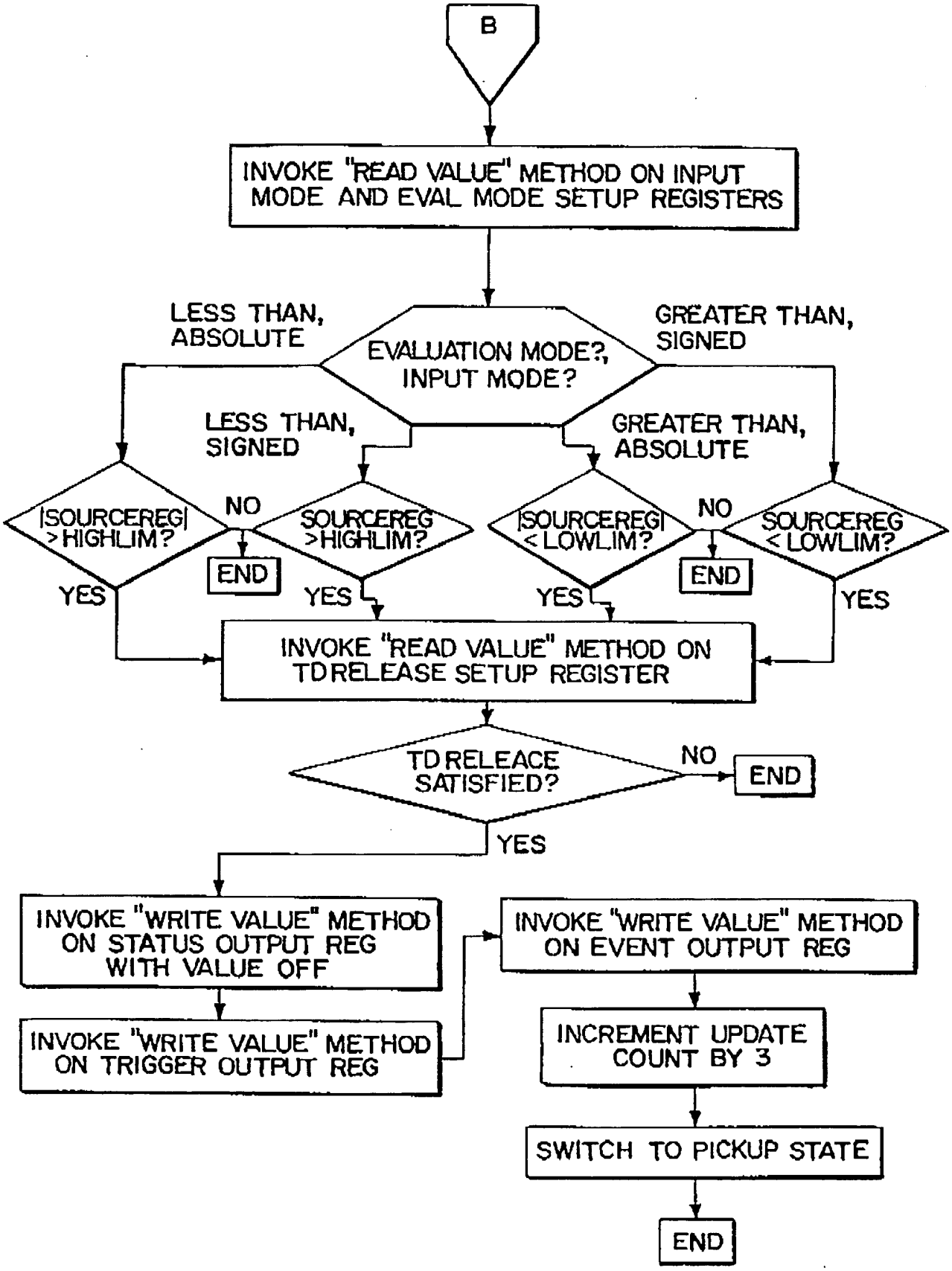

A
INVOKE "READ VALUE" METHOD ON INPUT AND EVAL MODE SETUP REGISTERS.
EVALUATION MODE?, INPUT MODE?
LESS THAN, ABSOLUTE
LESS THAN, SIGNED
GREATER THAN, ABSOLUTE
GREATER THAN, SIGNED
|SOURCE REG| <LOW LIM?
NO
END
YES
SOURCE REG <LOW LIM?
NO
END
YES
|SOURCE REG| >HIGH LIM?
NO
END
YES
SOURCE REG >HIGH LIM?
NO
END
YES
INVOKE "READ VALUE" METHOD ON TDOPERATE SETUP REGISTER
TDOPERATE SATISFIED?
NO
END
YES
INVOKE "WRITE VALUE" METHOD ON STATUS OUTPUT REG WITH VALVE ON
INVOKE "WRITE VALUE" METHOD ON TRIGGER OUTPUT REG
INVOKE "WRITE VALUE" METHOD ON EVENT OUTPUT REG
INCREMENT UPDATE COUNT BY 3
SWITCH TO ACTIVE STATE
END

FIG. 21C


B
INVOKE "READ VALUE" METHOD ON INPUT MODE AND EVAL MODE SETUP REGISTERS
LESS THAN, ABSOLUTE
EVALUATION MODE?, INPUT MODE?
GREATER THAN, SIGNED
LESS THAN, SIGNED
GREATER THAN, ABSOLUTE
|SOURCEREG| >HIGHLIM?
NO
SOURCEREG >HIGHLIM?
|SOURCEREG| <LOWLIM?
NO
SOURCEREG <LOWLIM?
END
END
YES
YES
YES
YES
INVOKE "READ VALUE" METHOD ON TDRELEASE SETUP REGISTER
TDRELEACE SATISFIED?
NO
END
YES
INVOKE "WRITE VALUE" METHOD ON STATUS OUTPUT REG WITH VALUE OFF
INVOKE "WRITE VALUE" METHOD ON TRIGGER OUTPUT REG
INVOKE "WRITE VALUE" METHOD ON EVENT OUTPUT REG
INCREMENT UPDATE COUNT BY 3
SWITCH TO PICKUP STATE
END

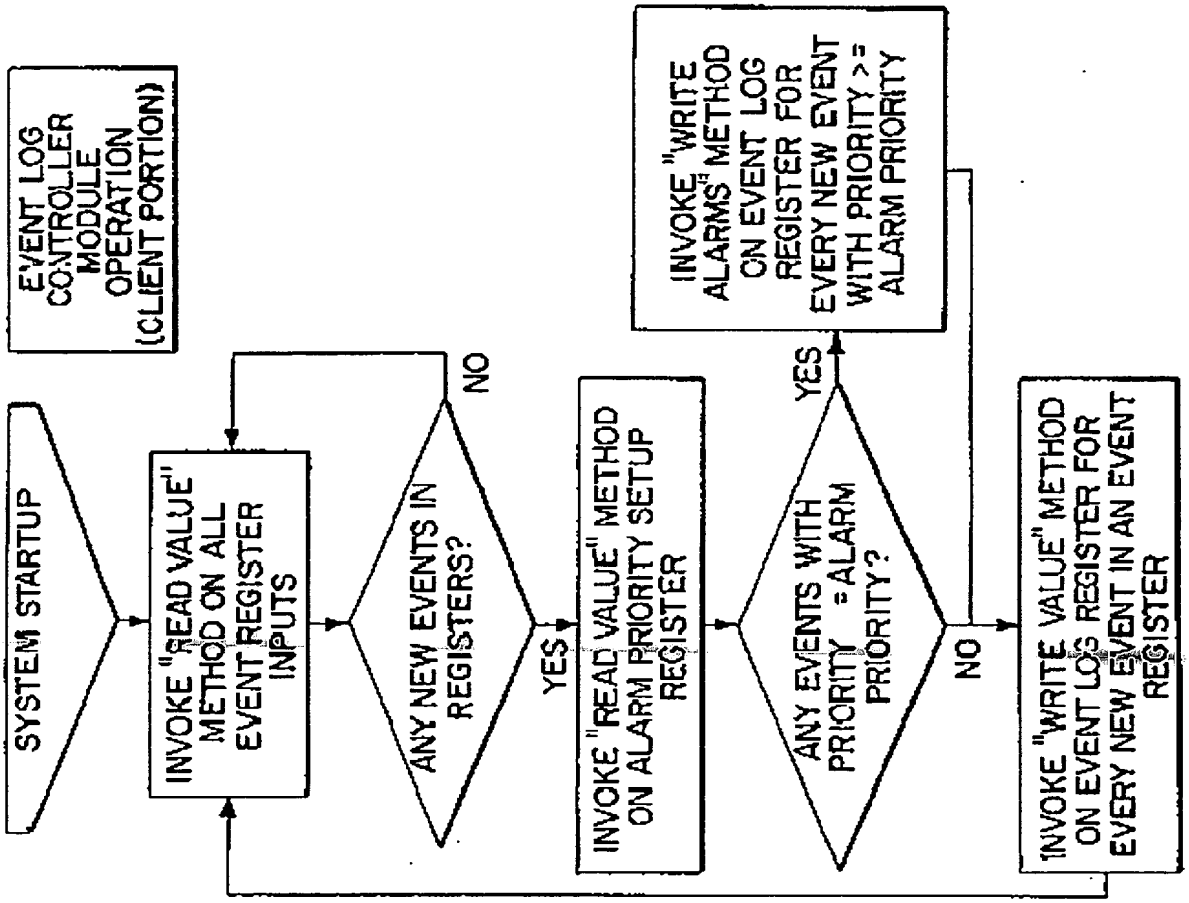


FIG. 22A

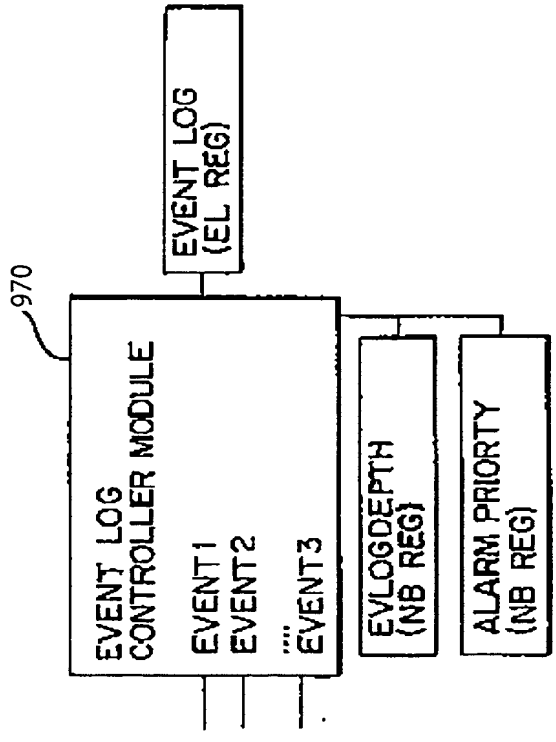


FIG. 22

FIG. 24A

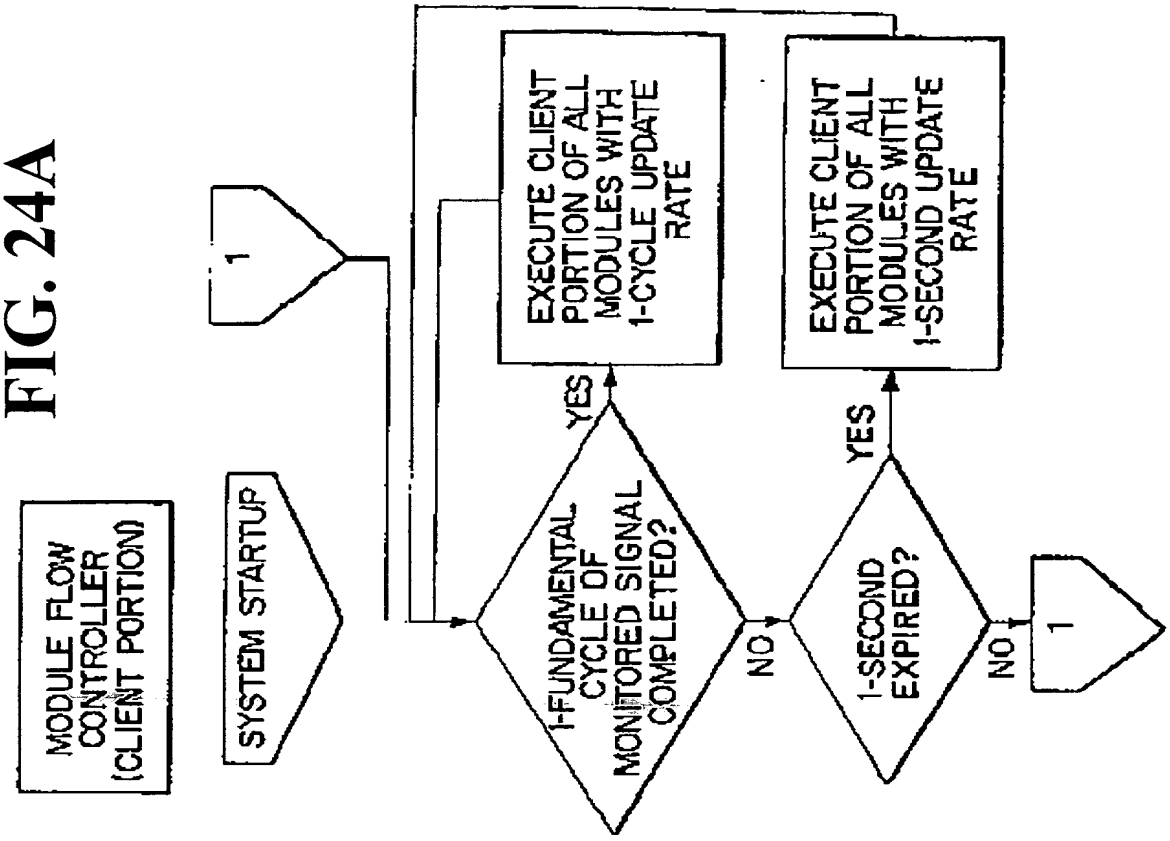


FIG. 23

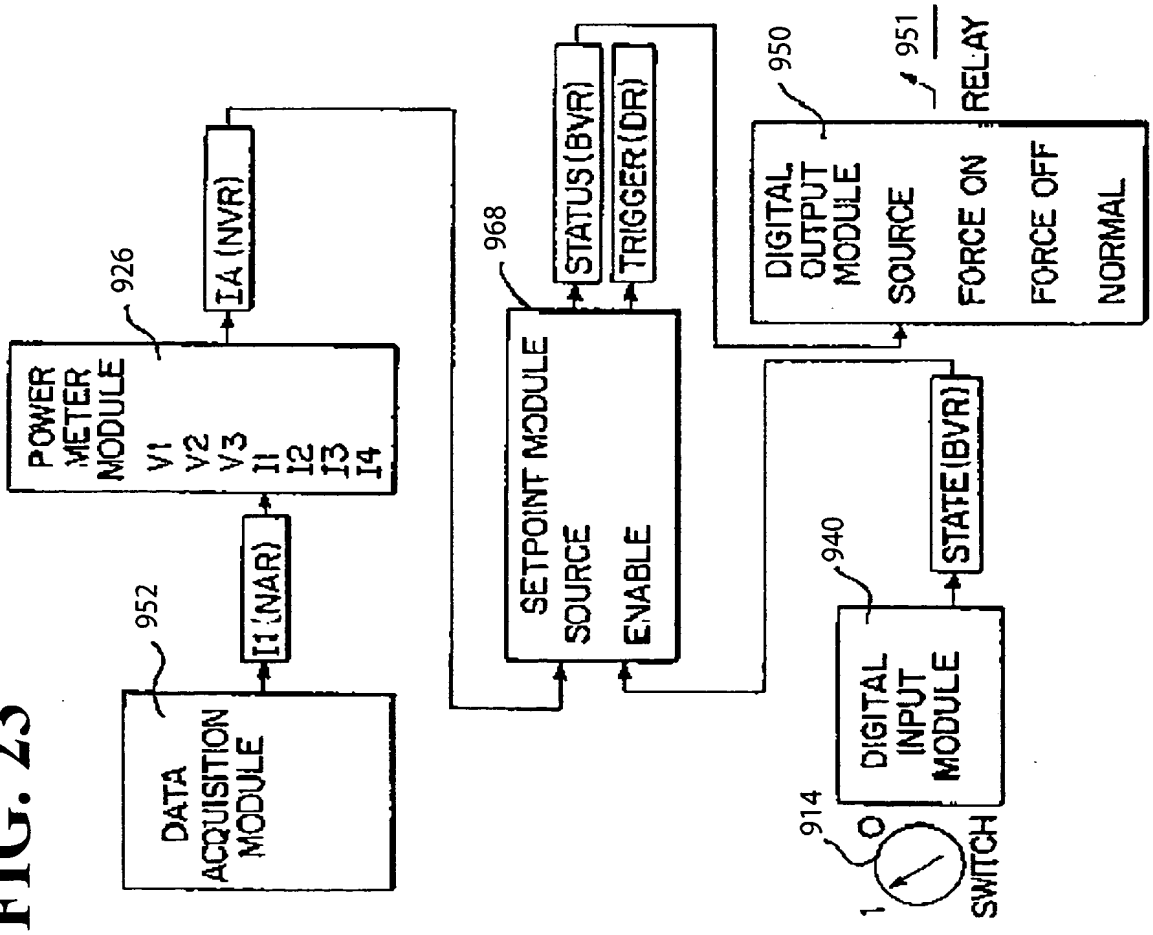


FIG. 25

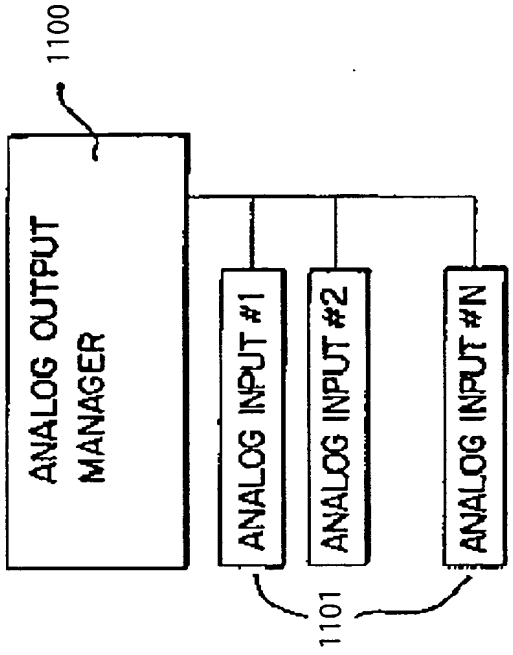


FIG. 24B

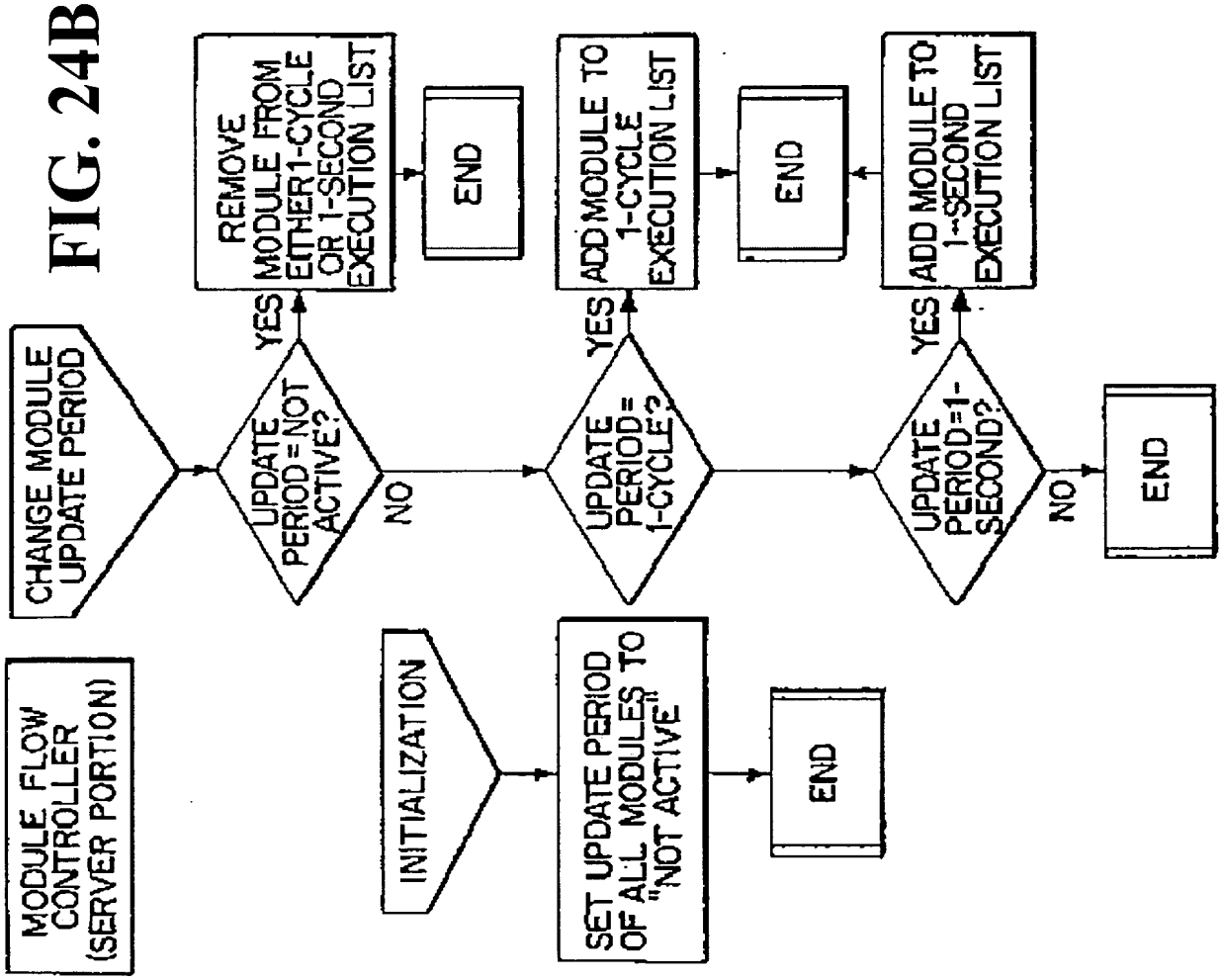


FIG. 25A

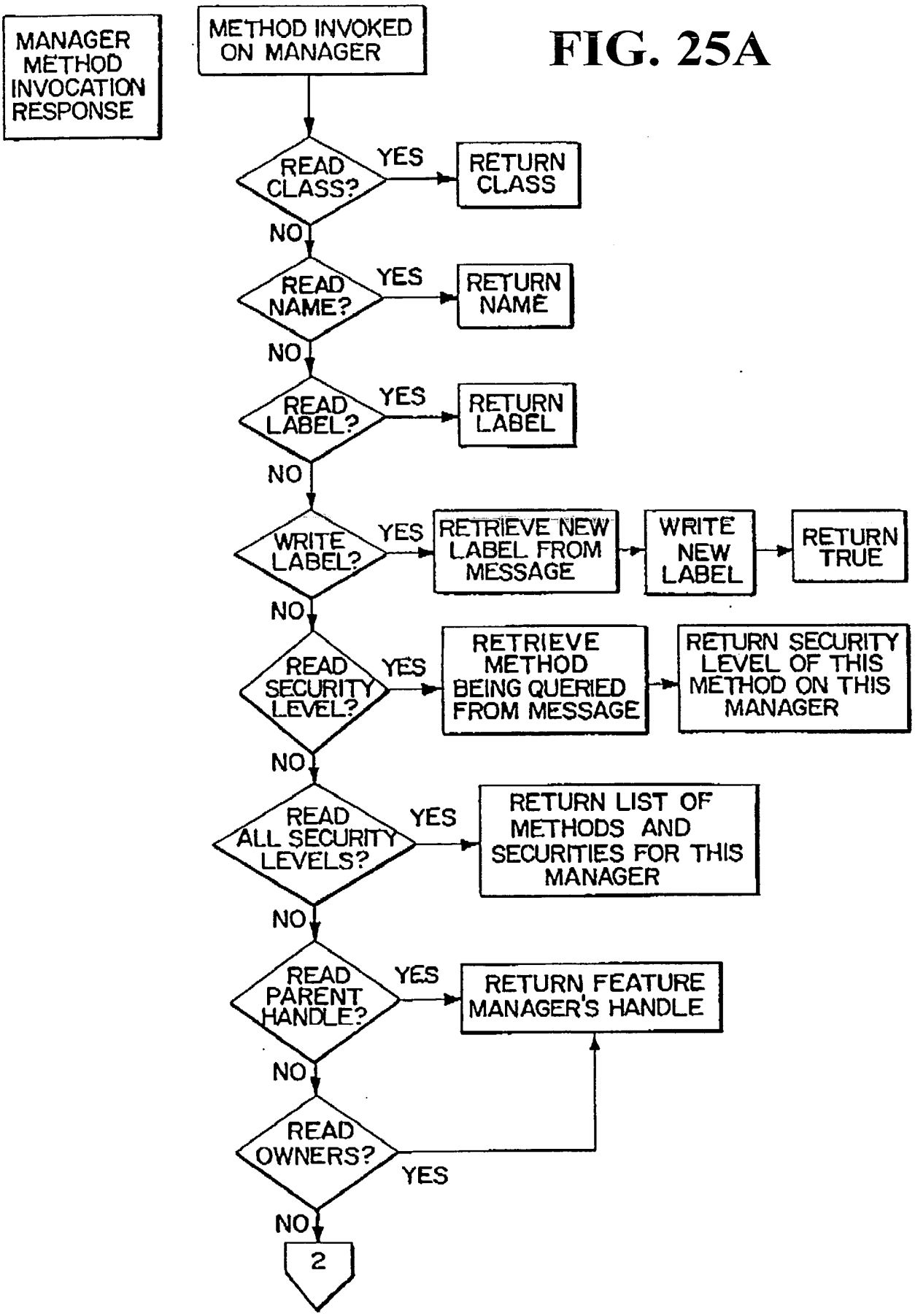

MANAGER METHOD INVOCATION RESPONSE
METHOD INVOKED ON MANAGER
READ CLASS?
YES
RETURN CLASS
NO
READ NAME?
YES
RETURN NAME
NO
READ LABEL?
YES
RETURN LABEL
NO
WRITE LABEL?
YES
RETRIEVE NEW LABEL FROM MESSAGE
WRITE NEW LABEL
RETURN TRUE
NO
READ SECURITY LEVEL?
YES
RETRIEVE METHOD BEING QUERIED FROM MESSAGE
RETURN SECURITY LEVEL OF THIS METHOD ON THIS MANAGER
NO
READ ALL SECURITY LEVELS?
YES
RETURN LIST OF METHODS AND SECURITIES FOR THIS MANAGER
NO
READ PARENT HANDLE?
YES
RETURN FEATURE MANAGER'S HANDLE
NO
READ OWNERS?
YES
NO
2

FIG. 25B

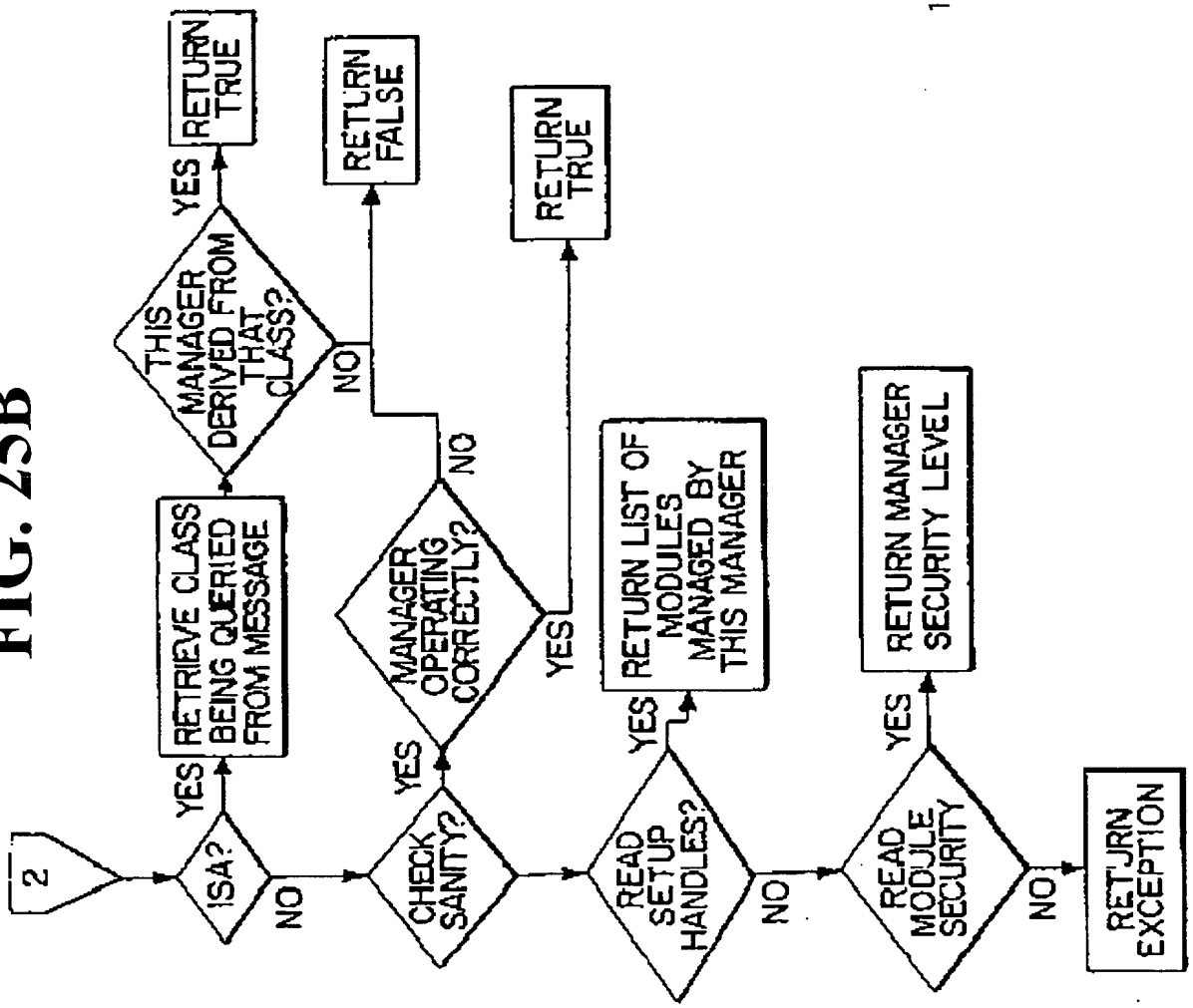


FIG. 26

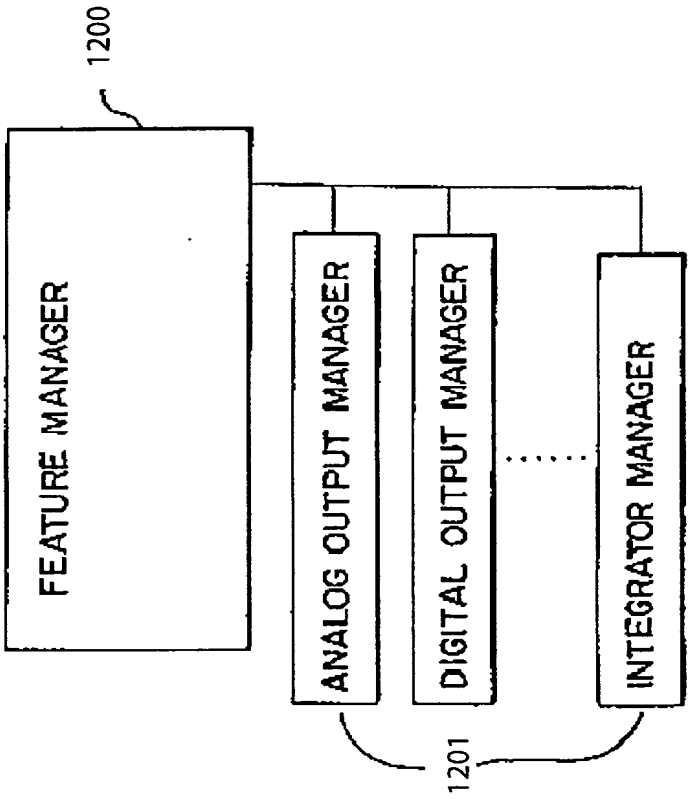


FIG. 26A

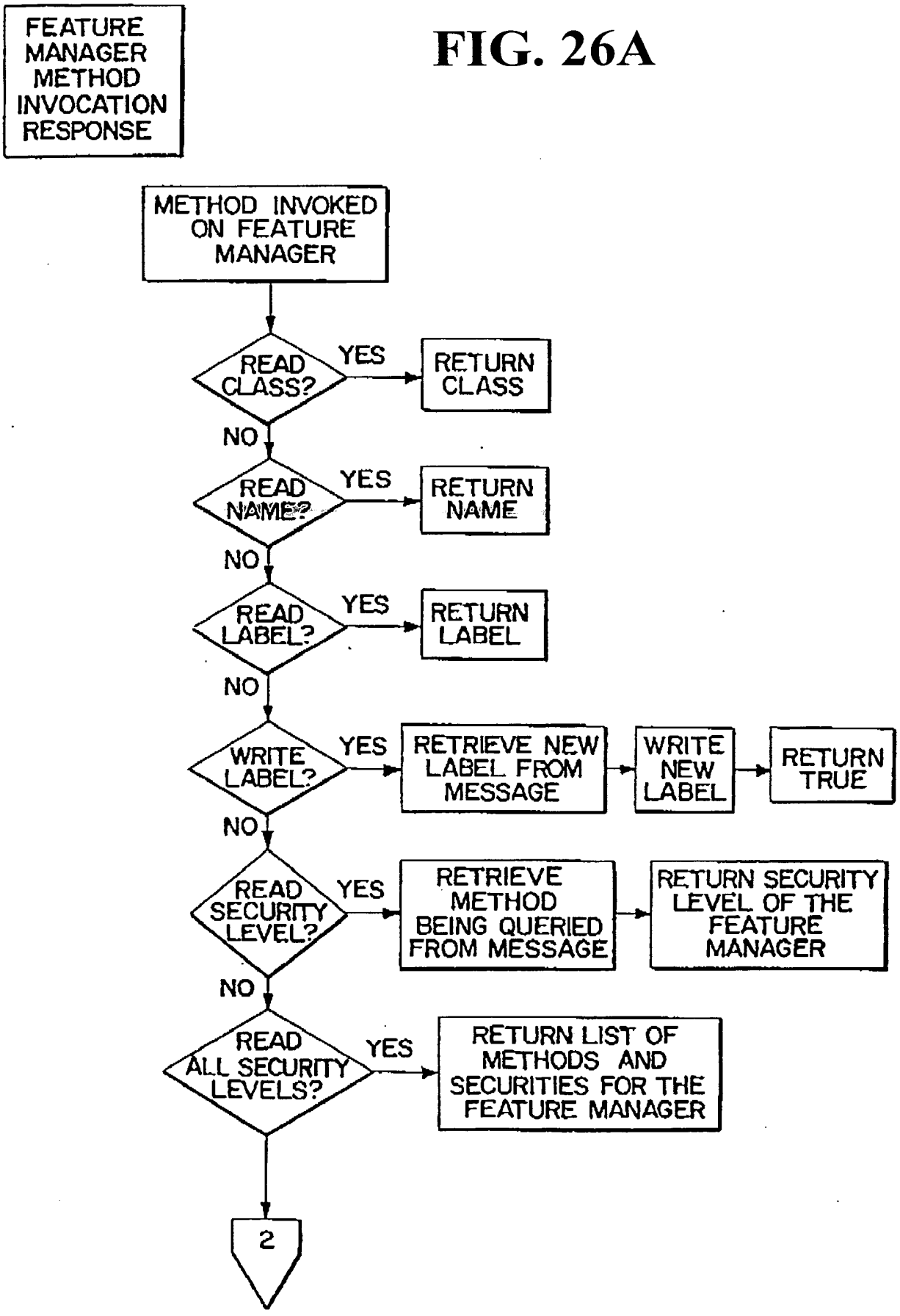
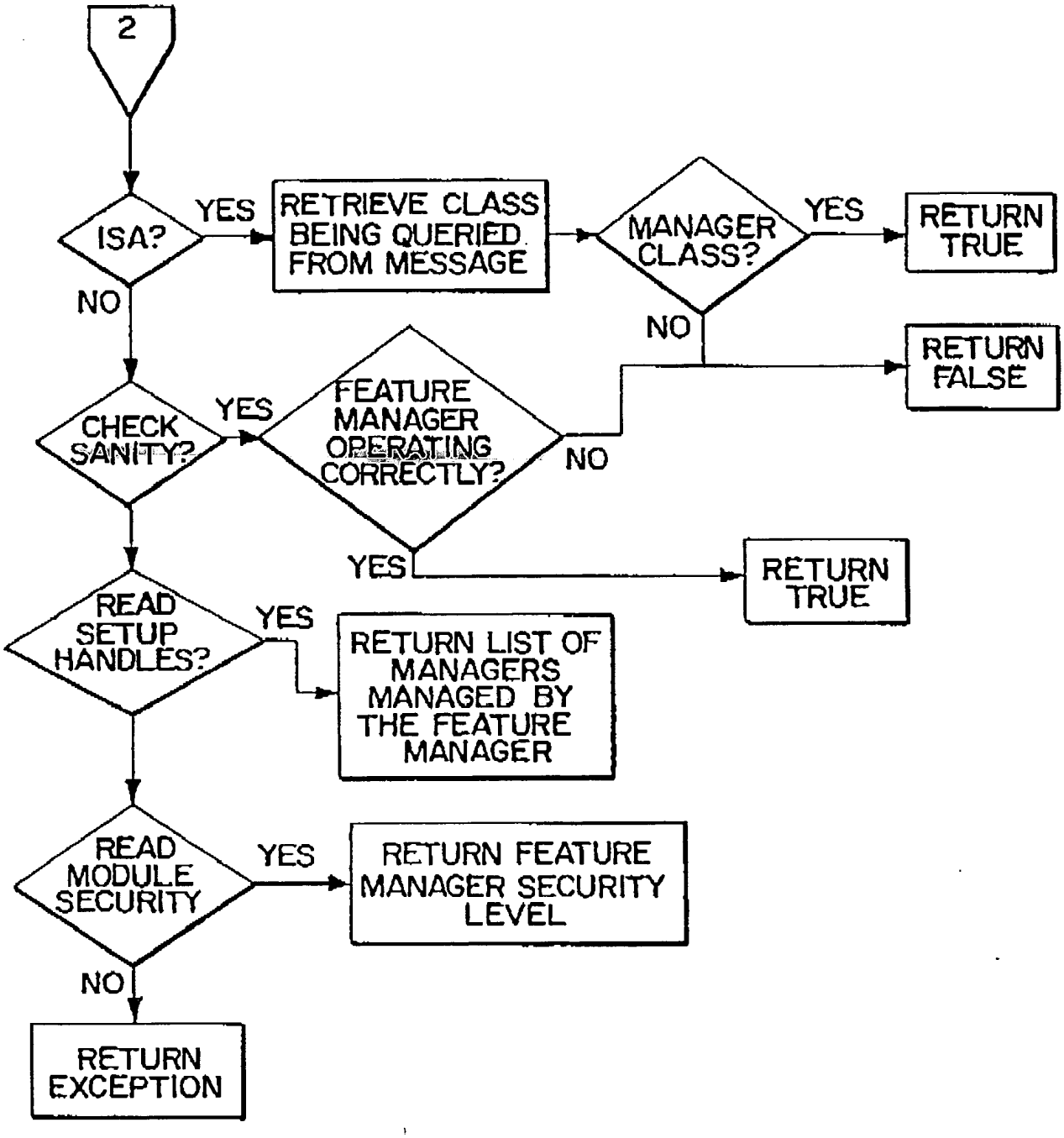

FEATURE MANAGER METHOD INVOCATION RESPONSE
METHOD INVOKED ON FEATURE MANAGER
READ CLASS?
YES
RETURN CLASS
NO
READ NAME?
YES
RETURN NAME
NO
READ LABEL?
YES
RETURN LABEL
NO
WRITE LABEL?
YES
RETRIEVE NEW LABEL FROM MESSAGE
WRITE NEW LABEL
RETURN TRUE
NO
READ SECURITY LEVEL?
YES
RETRIEVE METHOD BEING QUERIED FROM MESSAGE
RETURN SECURITY LEVEL OF THE FEATURE MANAGER
NO
READ ALL SECURITY LEVELS?
YES
RETURN LIST OF METHODS AND SECURITIES FOR THE FEATURE MANAGER
2

FIG. 26B


2
ISA?
YES
RETRIEVE CLASS BEING QUERIED FROM MESSAGE
MANAGER CLASS?
YES
RETURN TRUE
NO
NO
RETURN FALSE
CHECK SANITY?
YES
FEATURE MANAGER OPERATING CORRECTLY?
NO
YES
RETURN TRUE
READ SETUP HANDLES?
YES
RETURN LIST OF MANAGERS MANAGED BY THE FEATURE MANAGER
READ MODULE SECURITY
YES
RETURN FEATURE MANAGER SECURITY LEVEL
NO
RETURN EXCEPTION

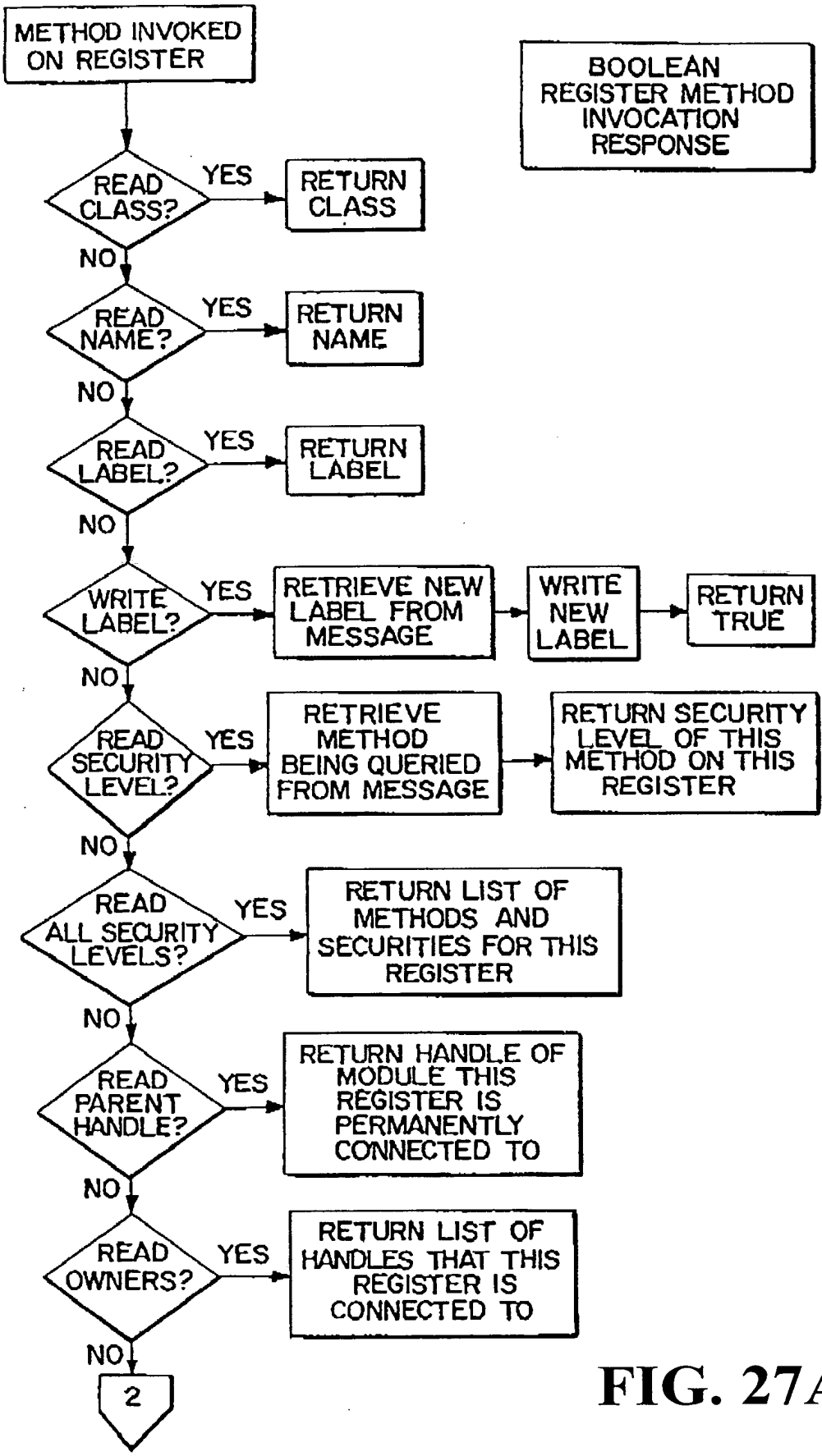

METHOD INVOKED ON REGISTER
BOOLEAN REGISTER METHOD INVOCATION RESPONSE
READ CLASS?
YES
RETURN CLASS
NO
READ NAME?
YES
RETURN NAME
NO
READ LABEL?
YES
RETURN LABEL
NO
WRITE LABEL?
YES
RETRIEVE NEW LABEL FROM MESSAGE
WRITE NEW LABEL
RETURN TRUE
NO
READ SECURITY LEVEL?
YES
RETRIEVE METHOD BEING QUERIED FROM MESSAGE
RETURN SECURITY LEVEL OF THIS METHOD ON THIS REGISTER
NO
READ ALL SECURITY LEVELS?
YES
RETURN LIST OF METHODS AND SECURITIES FOR THIS REGISTER
NO
READ PARENT HANDLE?
YES
RETURN HANDLE OF MODULE THIS REGISTER IS PERMANENTLY CONNECTED TO
NO
READ OWNERS?
YES
RETURN LIST OF HANDLES THAT THIS REGISTER IS CONNECTED TO
NO
2

FIG. 27A

FIG. 27B

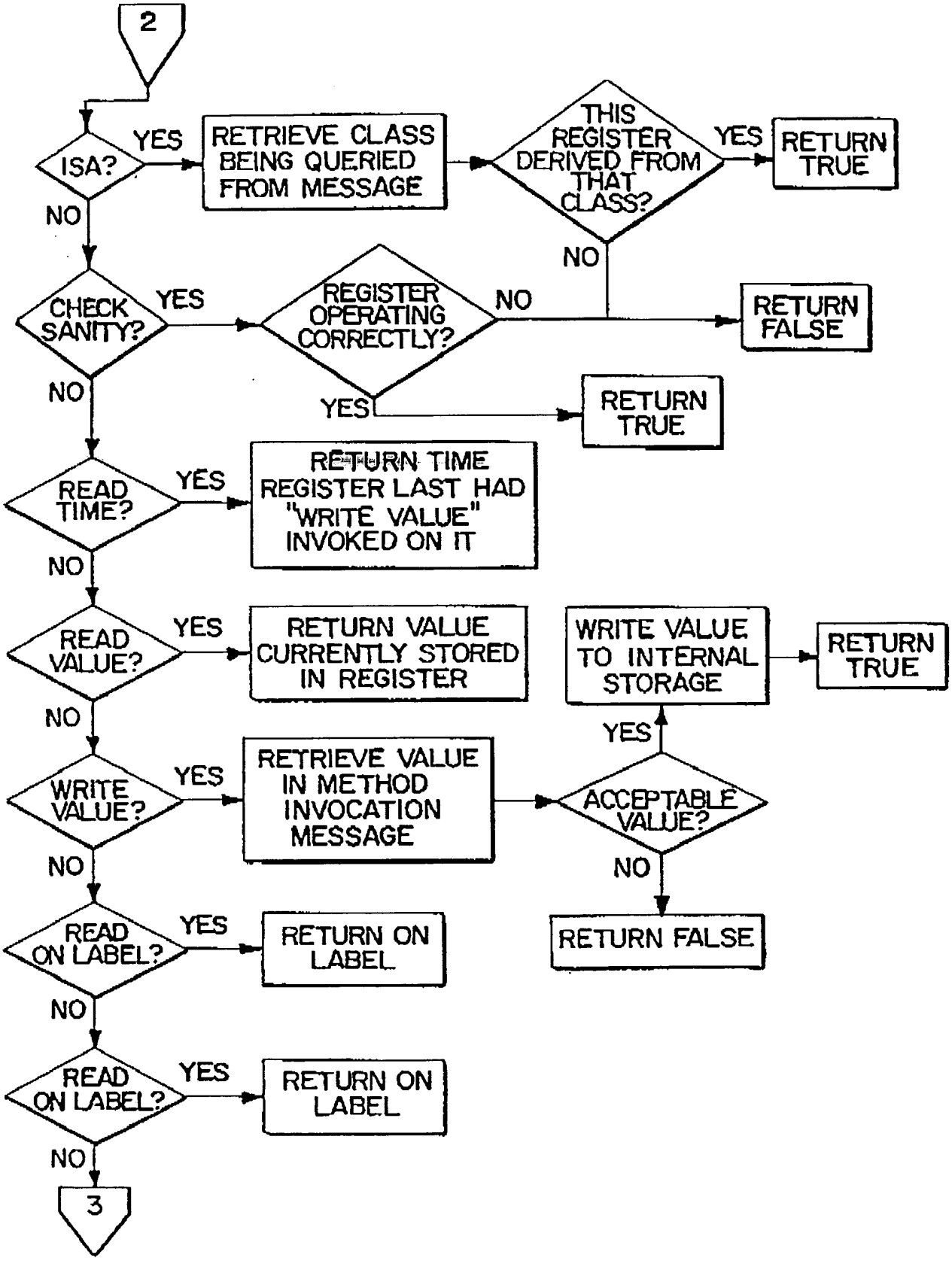

2
ISA?
YES
RETRIEVE CLASS BEING QUERIED FROM MESSAGE
THIS REGISTER DERIVED FROM THAT CLASS?
YES
RETURN TRUE
NO
NO
CHECK SANITY?
YES
REGISTER OPERATING CORRECTLY?
NO
RETURN FALSE
NO
YES
RETURN TRUE
READ TIME?
YES
RETURN TIME REGISTER LAST HAD "WRITE VALUE" INVOKED ON IT
NO
READ VALUE?
YES
RETURN VALUE CURRENTLY STORED IN REGISTER
WRITE VALUE TO INTERNAL STORAGE
RETURN TRUE
NO
YES
WRITE VALUE?
YES
RETRIEVE VALUE IN METHOD INVOCATION MESSAGE
ACCEPTABLE VALUE?
NO
NO
RETURN FALSE
READ ON LABEL?
YES
RETURN ON LABEL
NO
READ ON LABEL?
YES
RETURN ON LABEL
NO
3

FIG. 27C

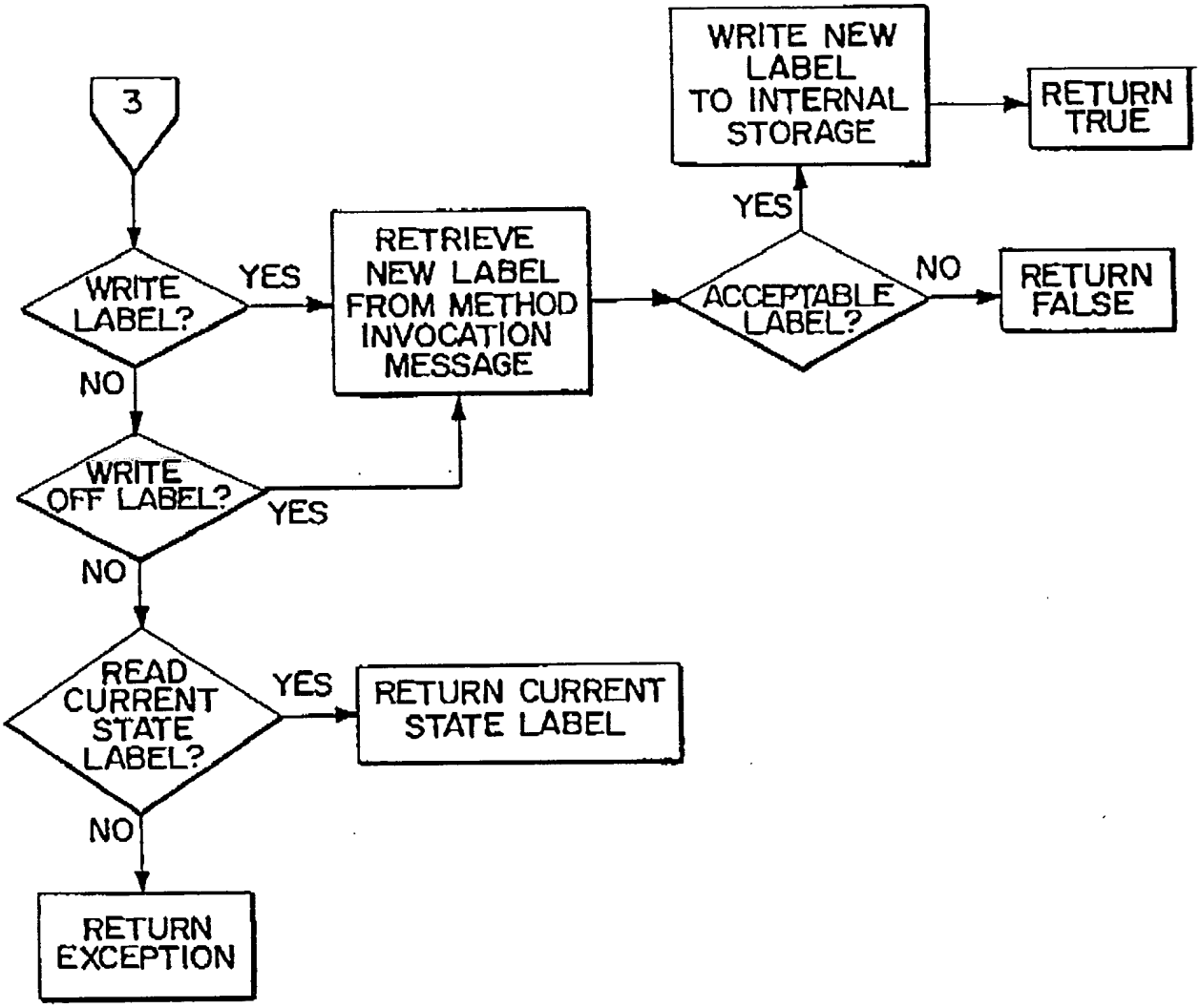

3
WRITE LABEL?
YES
RETRIEVE NEW LABEL FROM METHOD INVOCATION MESSAGE
NO
WRITE OFF LABEL?
YES
NO
READ CURRENT STATE LABEL?
YES
RETURN CURRENT STATE LABEL
NO
RETURN EXCEPTION
ACCEPTABLE LABEL?
NO
RETURN FALSE
YES
WRITE NEW LABEL TO INTERNAL STORAGE
RETURN TRUE

FIG. 28A

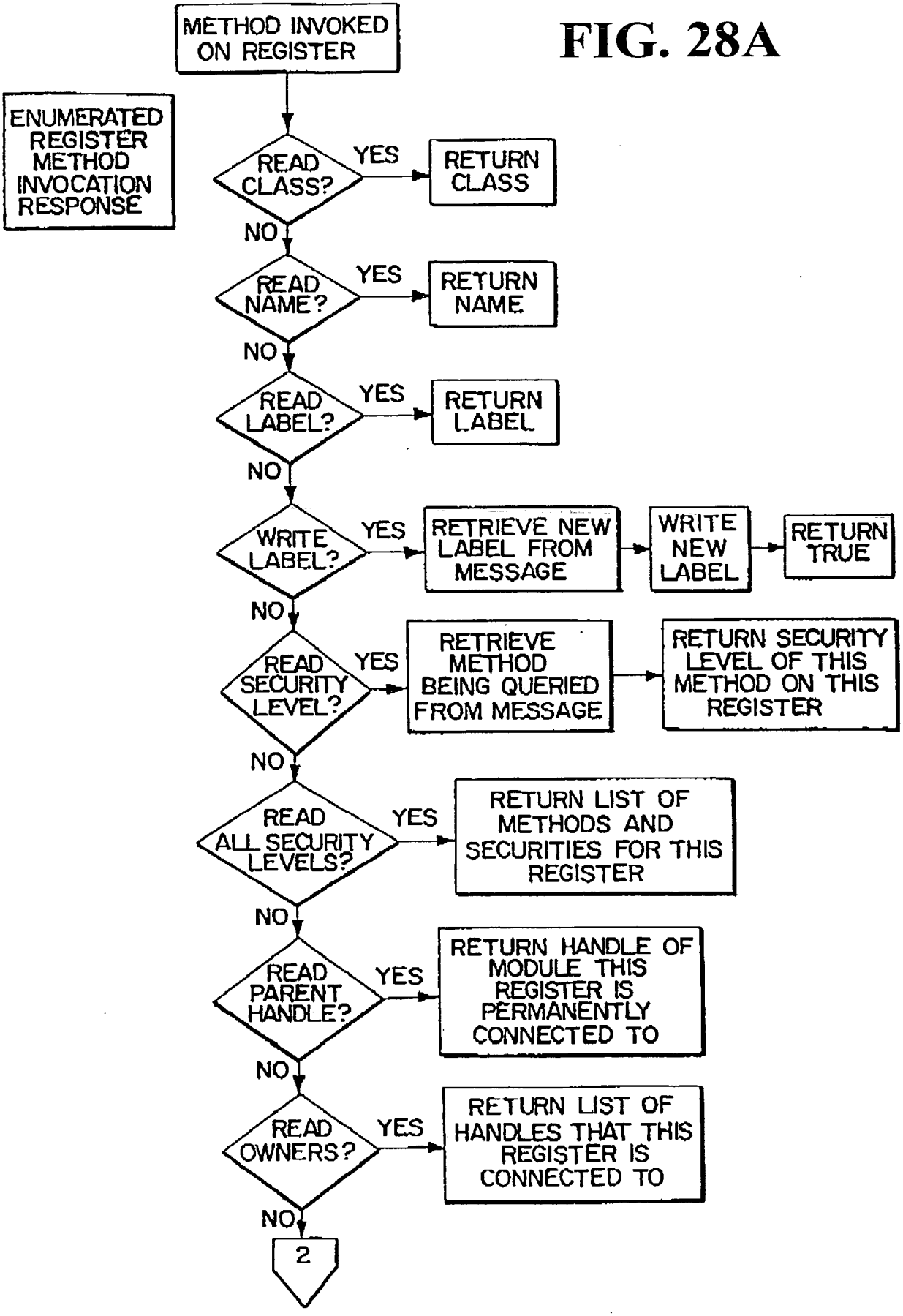

METHOD INVOKED ON REGISTER
ENUMERATED REGISTER METHOD INVOCATION RESPONSE
READ CLASS?
YES
RETURN CLASS
NO
READ NAME?
YES
RETURN NAME
NO
READ LABEL?
YES
RETURN LABEL
NO
WRITE LABEL?
YES
RETRIEVE NEW LABEL FROM MESSAGE
WRITE NEW LABEL
RETURN TRUE
NO
READ SECURITY LEVEL?
YES
RETRIEVE METHOD BEING QUERIED FROM MESSAGE
RETURN SECURITY LEVEL OF THIS METHOD ON THIS REGISTER
NO
READ ALL SECURITY LEVELS?
YES
RETURN LIST OF METHODS AND SECURITIES FOR THIS REGISTER
NO
READ PARENT HANDLE?
YES
RETURN HANDLE OF MODULE THIS REGISTER IS PERMANENTLY CONNECTED TO
NO
READ OWNERS?
YES
RETURN LIST OF HANDLES THAT THIS REGISTER IS CONNECTED TO
NO
2

FIG. 28B

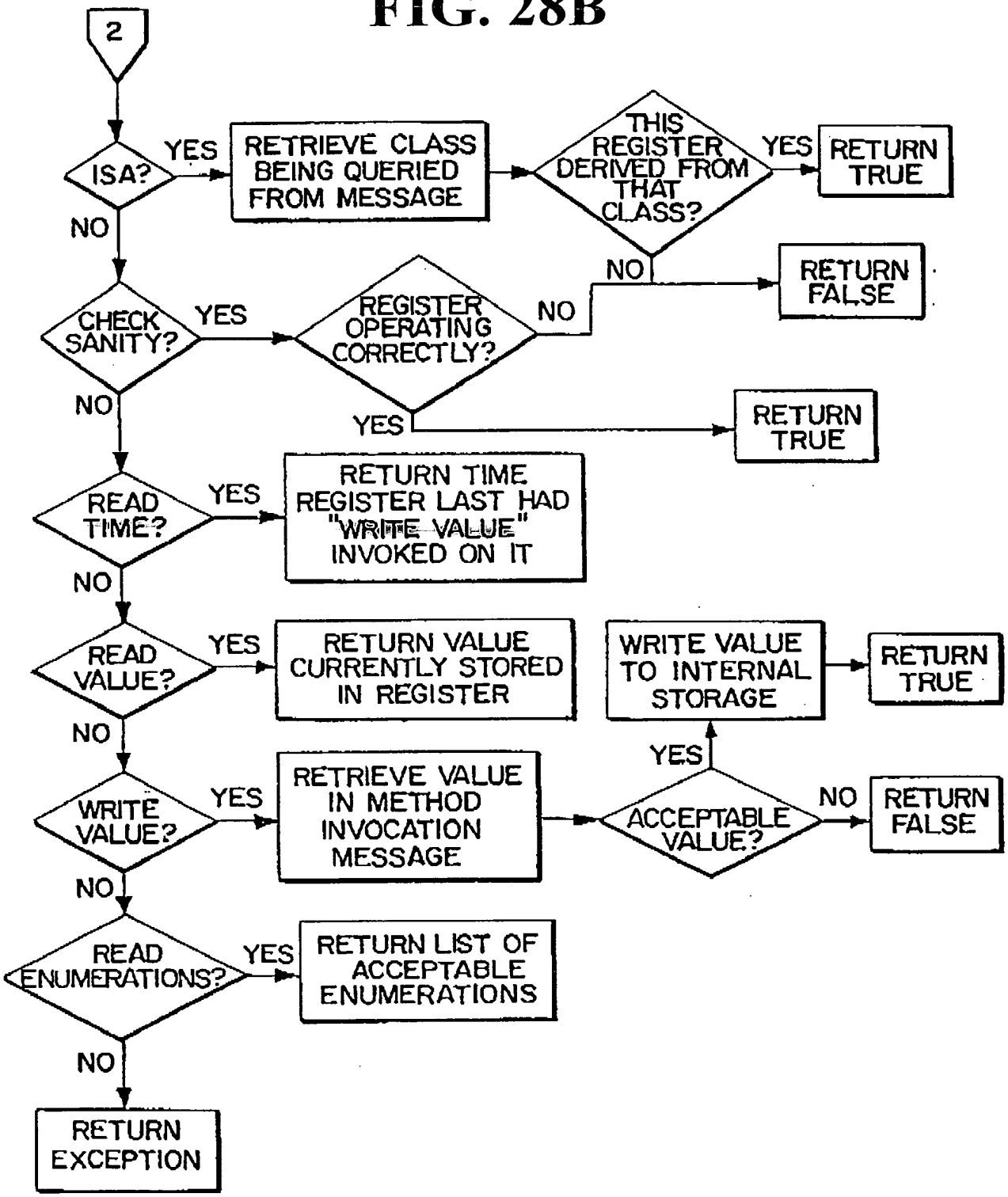

2
ISA?
YES
RETRIEVE CLASS BEING QUERIED FROM MESSAGE
THIS REGISTER DERIVED FROM THAT CLASS?
YES
RETURN TRUE
NO
RETURN FALSE
NO
CHECK SANITY?
YES
REGISTER OPERATING CORRECTLY?
NO
YES
RETURN TRUE
NO
READ TIME?
YES
RETURN TIME REGISTER LAST HAD "WRITE VALUE" INVOKED ON IT
NO
READ VALUE?
YES
RETURN VALUE CURRENTLY STORED IN REGISTER
WRITE VALUE TO INTERNAL STORAGE
RETURN TRUE
NO
YES
WRITE VALUE?
YES
RETRIEVE VALUE IN METHOD INVOCATION MESSAGE
ACCEPTABLE VALUE?
NO
RETURN FALSE
NO
READ ENUMERATIONS?
YES
RETURN LIST OF ACCEPTABLE ENUMERATIONS
NO
RETURN EXCEPTION

FIG. 29A

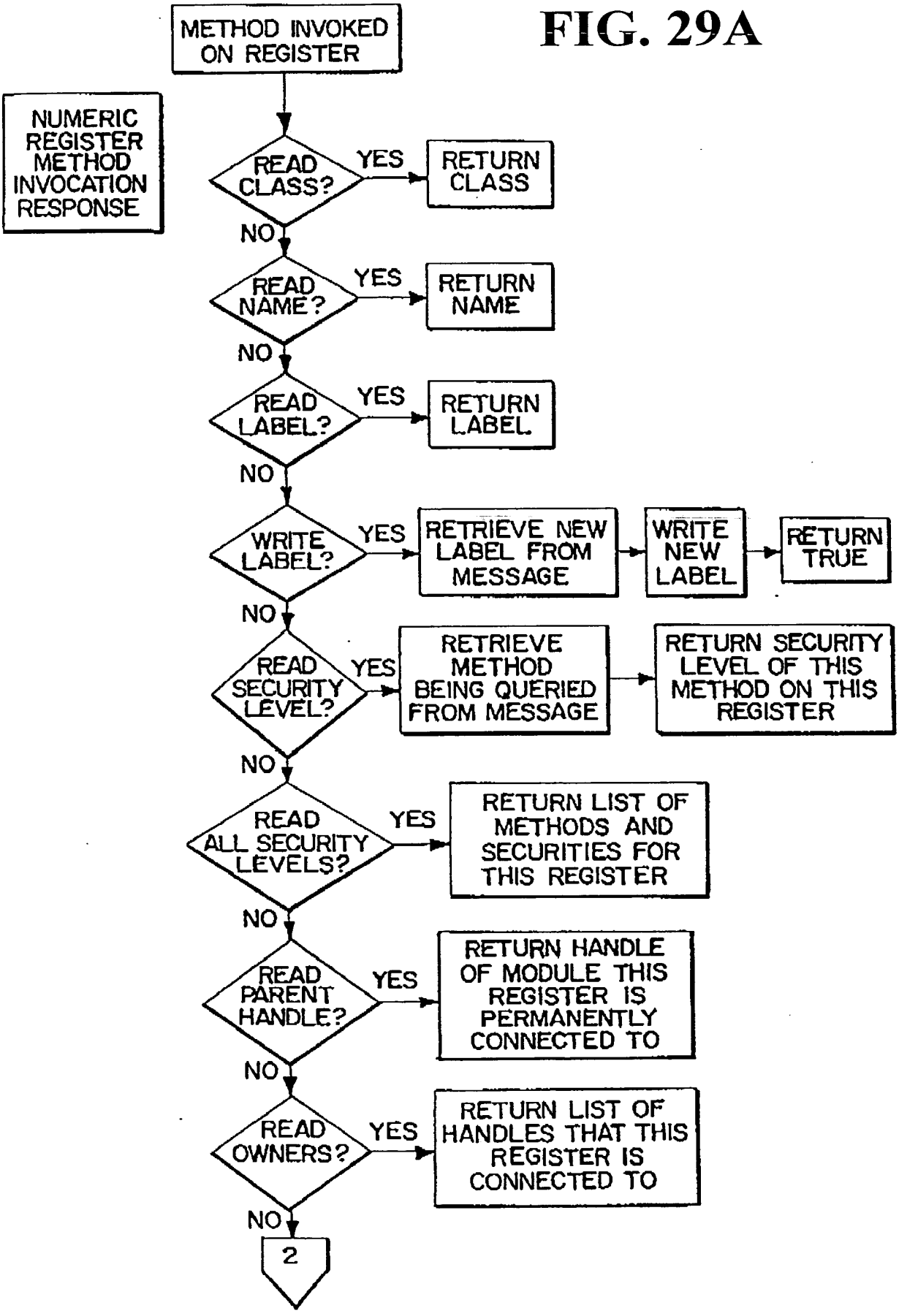

METHOD INVOKED ON REGISTER
NUMERIC REGISTER METHOD INVOCATION RESPONSE
READ CLASS?
YES
RETURN CLASS
NO
READ NAME?
YES
RETURN NAME
NO
READ LABEL?
YES
RETURN LABEL
NO
WRITE LABEL?
YES
RETRIEVE NEW LABEL FROM MESSAGE
WRITE NEW LABEL
RETURN TRUE
NO
READ SECURITY LEVEL?
YES
RETRIEVE METHOD BEING QUERIED FROM MESSAGE
RETURN SECURITY LEVEL OF THIS METHOD ON THIS REGISTER
NO
READ ALL SECURITY LEVELS?
YES
RETURN LIST OF METHODS AND SECURITIES FOR THIS REGISTER
NO
READ PARENT HANDLE?
YES
RETURN HANDLE OF MODULE THIS REGISTER IS PERMANENTLY CONNECTED TO
NO
READ OWNERS?
YES
RETURN LIST OF HANDLES THAT THIS REGISTER IS CONNECTED TO
NO
2

FIG. 29B

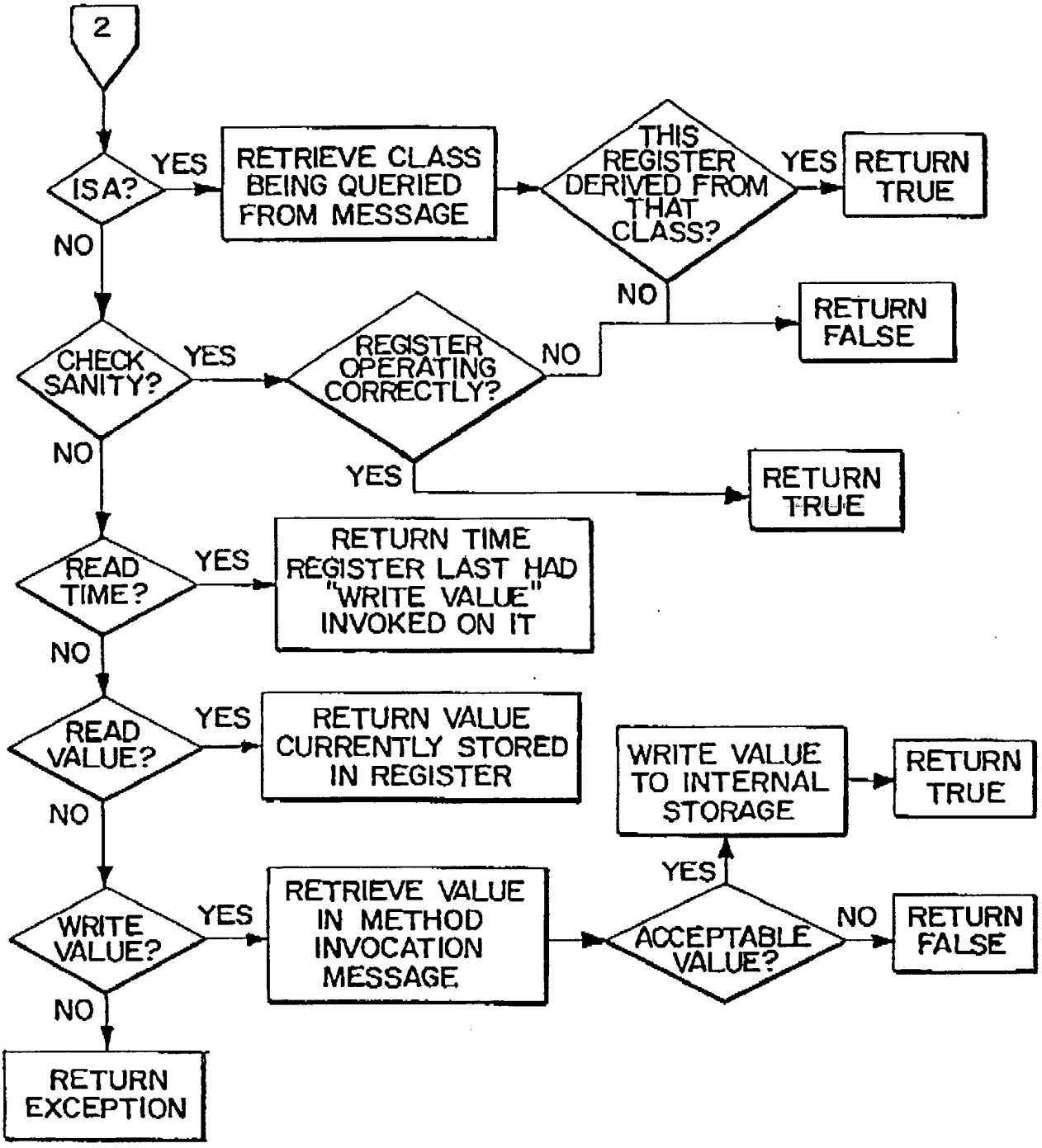

2
ISA?
YES
RETRIEVE CLASS BEING QUERIED FROM MESSAGE
THIS REGISTER DERIVED FROM THAT CLASS?
YES
RETURN TRUE
NO
NO
RETURN FALSE
CHECK SANITY?
YES
REGISTER OPERATING CORRECTLY?
NO
NO
YES
RETURN TRUE
READ TIME?
YES
RETURN TIME REGISTER LAST HAD "WRITE VALUE" INVOKED ON IT
NO
READ VALUE?
YES
RETURN VALUE CURRENTLY STORED IN REGISTER
NO
WRITE VALUE TO INTERNAL STORAGE
RETURN TRUE
YES
WRITE VALUE?
YES
RETRIEVE VALUE IN METHOD INVOCATION MESSAGE
ACCEPTABLE VALUE?
NO
RETURN FALSE
NO
RETURN EXCEPTION

FIG. 30A

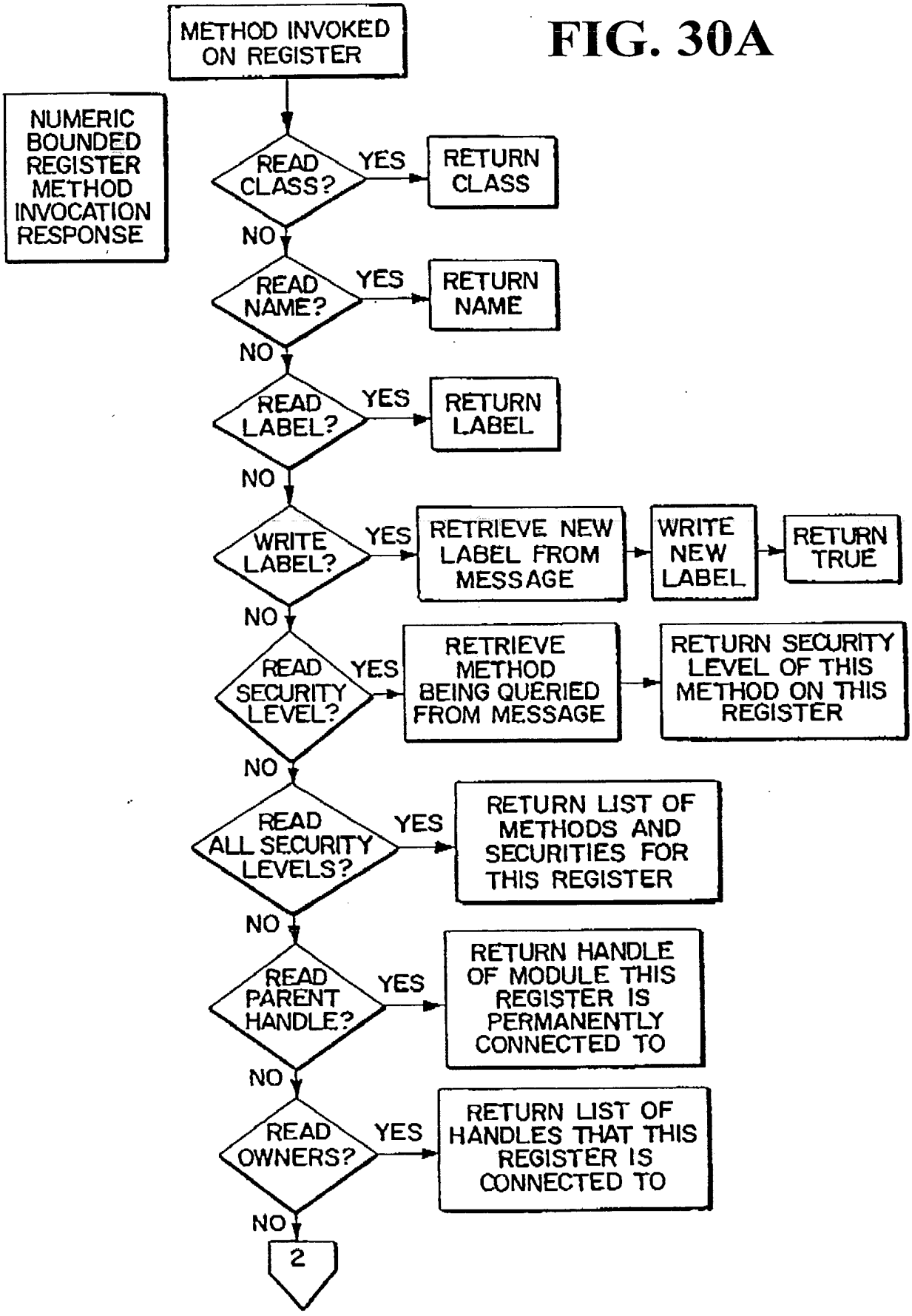
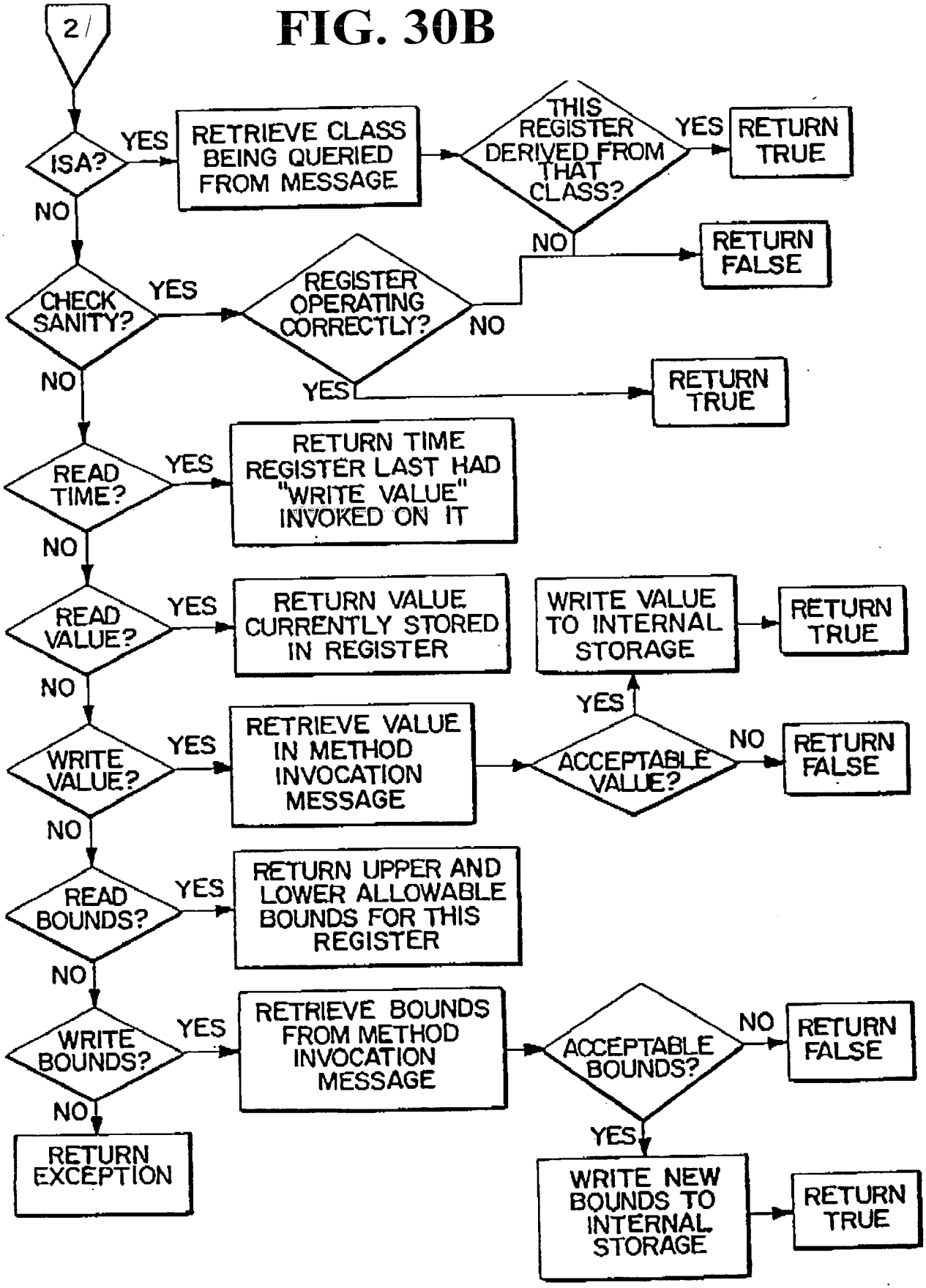

METHOD INVOKED ON REGISTER
NUMERIC BOUNDED REGISTER METHOD INVOCATION RESPONSE
READ CLASS?
YES
RETURN CLASS
NO
READ NAME?
YES
RETURN NAME
NO
READ LABEL?
YES
RETURN LABEL
NO
WRITE LABEL?
YES
RETRIEVE NEW LABEL FROM MESSAGE
WRITE NEW LABEL
RETURN TRUE
NO
READ SECURITY LEVEL?
YES
RETRIEVE METHOD BEING QUERIED FROM MESSAGE
RETURN SECURITY LEVEL OF THIS METHOD ON THIS REGISTER
NO
READ ALL SECURITY LEVELS?
YES
RETURN LIST OF METHODS AND SECURITIES FOR THIS REGISTER
NO
READ PARENT HANDLE?
YES
RETURN HANDLE OF MODULE THIS REGISTER IS PERMANENTLY CONNECTED TO
NO
READ OWNERS?
YES
RETURN LIST OF HANDLES THAT THIS REGISTER IS CONNECTED TO
NO
2

FIG. 30B


2/
ISA?
YES
RETRIEVE CLASS BEING QUERIED FROM MESSAGE
THIS REGISTER DERIVED FROM THAT CLASS?
YES
RETURN TRUE
NO
RETURN FALSE
NO
CHECK SANITY?
YES
REGISTER OPERATING CORRECTLY?
NO
YES
RETURN TRUE
NO
READ TIME?
YES
RETURN TIME REGISTER LAST HAD "WRITE VALUE" INVOKED ON IT
NO
READ VALUE?
YES
RETURN VALUE CURRENTLY STORED IN REGISTER
WRITE VALUE TO INTERNAL STORAGE
RETURN TRUE
NO
WRITE VALUE?
YES
RETRIEVE VALUE IN METHOD INVOCATION MESSAGE
ACCEPTABLE VALUE?
YES
NO
RETURN FALSE
NO
READ BOUNDS?
YES
RETURN UPPER AND LOWER ALLOWABLE BOUNDS FOR THIS REGISTER
NO
WRITE BOUNDS?
YES
RETRIEVE BOUNDS FROM METHOD INVOCATION MESSAGE
ACCEPTABLE BOUNDS?
NO
RETURN FALSE
YES
NO
RETURN EXCEPTION
WRITE NEW BOUNDS TO INTERNAL STORAGE
RETURN TRUE